Use of microcomputers in mathematics in Hong Kong higher education

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/19483

Publisher: © T.Y.G. Pong

Rights: This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) licence. Full details of this licence are available at: https://creativecommons.org/licenses/by-nc-nd/4.0/

Please cite the published version.
USE OF MICROCOMPUTERS IN MATHEMATICS
IN HONG KONG HIGHER EDUCATION

by
Tak-Yun G Pong
BSc MSc

A doctoral thesis submitted in partial fulfillment of the requirements for the award of Doctor of Philosophy of the Loughborough University of Technology, 1988

Supervisor : Professor A C Bajpai OBE, CEng, Companion IEE
Head of Department of Engineering Mathematics
Loughborough University of Technology

© by T Y G Pong, 1988
ABSTRACT

Use of Microcomputers in Mathematics in Hong Kong Higher Education

by Tak-Yun G Pong

CAMET
(Centre for Advancement of Mathematical Education in Technology)
Loughborough University of Technology

Since the innovation of computers some 40 years ago and the introduction of microcomputers in 1975, computers are playing an active role in education processes and altering the pattern of interaction between teacher and student in the classroom. Computer assisted learning has been seen as a revolution in education. In this research, the author has studied the impact of using microcomputers on mathematical education, particularly at the Hong Kong tertiary level, in different perspectives.
Two computer software packages have been developed on the microcomputer. The consideration of the topic to be used in the computer assisted learning was arrived at in earlier surveys with students who thought that computers could give very accurate solutions to calculations. The two software packages, demonstrating on the spot the error that would be incurred by the computer, have been used by the students. They are both interactive and make use of the advantages of the microcomputer's functions over other teaching media, such as graphics facility and random number generator, to draw to the students' attention awareness of errors that may be obtained using computers in numerical solutions.

Much emphasis is put on the significance and effectiveness of using computer packages in learning and teaching. Measurements are based on questionnaires, conversations with students, and tests on content material after the packages have been used. Feedback and subjective opinion of using computers in mathematical education have also been obtained from both students and other teachers.

The research then attempts to examine the suitability of applying computer assisted learning in Hong Kong education sectors. Some studies on the comments made by students who participated in the learning process are undertaken. The successes and failures in terms of student accomplishment and interest in the subject area as a result of using a software package is described. Suggestions and recommendations are given in the concluding chapter.
Key words: Computer Assisted Learning, Computer Enhanced Learning, Microcomputers in Mathematics, Rounding Error, Truncation Error, Numerical Methods
ACKNOWLEDGMENTS

I am indebted to my supervisor, Professor A.C. Bajpai, OBE for all his valuable advice and encouragement in guiding me through the entire research project. I gratefully acknowledge my Head of Department Dr. I. Tang for his help and understanding, particularly for all the time I had to devote to work on the thesis; and the Staff Development Sub-Committee of Hong Kong Polytechnic for its financial support. Also, I wish to express my sincere thanks to all my colleagues, especially Dr. A. Lun, and the students who gave suggestions and comments and took part in the evaluations, their names are too numerous to list.

Finally, I am deeply indebted to my family, Aster, Vivien and Ian, for their continuous support.
# CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1 The Introduction</strong></td>
<td></td>
</tr>
<tr>
<td>1.1 Historical Background</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Author’s Background</td>
<td>3</td>
</tr>
<tr>
<td>1.3 The Scope of the Thesis</td>
<td>4</td>
</tr>
<tr>
<td><strong>2 The Computer Era</strong></td>
<td></td>
</tr>
<tr>
<td>2.1 Yesterday, Today, and Tomorrow</td>
<td>9</td>
</tr>
<tr>
<td>2.2 Hong Kong in General</td>
<td>12</td>
</tr>
<tr>
<td>2.3 The Secondary Schools</td>
<td>13</td>
</tr>
<tr>
<td>2.3.1 The Computer Studies Pilot Scheme</td>
<td>14</td>
</tr>
<tr>
<td>2.3.2 Exceptional Cases</td>
<td>15</td>
</tr>
<tr>
<td>2.4 The Polytechnics in Hong Kong</td>
<td>18</td>
</tr>
<tr>
<td>2.4.1 The Hong Kong Polytechnic</td>
<td>18</td>
</tr>
<tr>
<td>2.4.2 The City Polytechnic</td>
<td>20</td>
</tr>
<tr>
<td>2.5 The Universities</td>
<td>21</td>
</tr>
<tr>
<td><strong>3 Computer Literacy In Hong Kong Education Sectors</strong></td>
<td></td>
</tr>
<tr>
<td>3.1 The Secondary Schools</td>
<td>23</td>
</tr>
<tr>
<td>3.1.1 The Objectives and Syllabus</td>
<td>24</td>
</tr>
<tr>
<td>3.1.2 The Extra-Curriculum</td>
<td>26</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>3.1.3</td>
<td>Feedbacks</td>
</tr>
<tr>
<td>3.1.4</td>
<td>Further Expectations</td>
</tr>
<tr>
<td>3.2</td>
<td>The Polytechnics</td>
</tr>
<tr>
<td>3.3</td>
<td>The Universities</td>
</tr>
<tr>
<td>3.4</td>
<td>Follow-up Training</td>
</tr>
<tr>
<td>4</td>
<td>Survey On Computer Awareness Of HKP Students</td>
</tr>
<tr>
<td>4.1</td>
<td>Why the Survey</td>
</tr>
<tr>
<td>4.2</td>
<td>The Surveys</td>
</tr>
<tr>
<td>4.3</td>
<td>The Target Population and the Sample</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Tabulation Results on Closed-Answer Questions</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Cross-Tabulation Results</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Analysis on Self-Judgement Questions</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Computer Ideas Questions</td>
</tr>
<tr>
<td>4.4.5</td>
<td>Their Computer World</td>
</tr>
<tr>
<td>4.5</td>
<td>Conclusion</td>
</tr>
<tr>
<td>5</td>
<td>A CAL Package Used By Students Of HKP</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
</tr>
<tr>
<td>5.2</td>
<td>The Aim and Content</td>
</tr>
<tr>
<td>5.3</td>
<td>The Approach</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Authoring Language</td>
</tr>
<tr>
<td>5.3.2</td>
<td>The Design and Layout</td>
</tr>
<tr>
<td>5.3.3</td>
<td>The Coding</td>
</tr>
<tr>
<td>5.4</td>
<td>The Use</td>
</tr>
</tbody>
</table>
6 Feedback From The CAL Lesson
   6.1 The Question Paper and the Evaluation 101
   6.2 Lesson Content Questions 106
   6.3 Questions on Student's Particulars 112
   6.4 Feedback Questions 117
   6.5 Open Answer Question 125
   6.6 Summary 128

7 Computer Assisted Learning
   7.1 Introduction 134
   7.2 The PLATO Project 136
   7.3 The MIME Project 140
   7.4 Reasons for CAL 146
   7.5 Computer Enhanced Learning (CEL) 150
   7.6 Summary 152

8 A Second CAL Package Used By Students Of HKP
   8.1 Introduction 156
   8.2 The Aim and Content 161
   8.3 The Approach 166
   8.3.1 Authoring System 168
      8.3.1.1 Domino 168
      8.3.1.2 MHIAS 170
   8.3.2 General Computer Language 173
   8.3.3 The Coding 175
   8.4 The Use 179
9 Feedback From The Second CAL Lesson

9.1 The Question Paper and the Evaluation 181
9.2 Lesson Content Questions 184
9.3 Questions on Student’s Particulars 191
9.4 Feedback Questions 198
9.5 Open Answer Question 209
9.6 Summary 210

10 Summary, Recommendations and Suggestions for Further Research

10.1 Comparison of the Two Evaluations 217
10.2 Summary 219
10.3 Recommendations and Suggestions 222

for Further Research

REFERENCES

APPENDICES
CHAPTER 1

The Introduction

1.1 Historical Background

The earliest computer is believed to be the abacus used in China more than two thousand years ago. It was then not until the middle of this century that the first electronic digital computer was invented. But since this initial invention, microelectronic technology has grown so rapidly that new ideas, new products and new concepts appear daily: the mainframe computer, the minicomputer, and then in the 1980s, the microcomputer. The microcomputer is not a cheap alternative to a large scale computer as far as instruction to the computer is concerned. It is true that the mainframe computer runs a lot faster, can have a massive amount of storage space, and probably has a more sophisticated I/O (Input/Output) structure. The microcomputer is simply an implementation of portions of a computer. The amount of implementation happens to be small; but even so, parts of the computer are still present. Microcomputers also have the advantage of their powerful graphics capability. Due to the price drop of electronic hardware, the use of
microcomputers in the education sector has become widespread.

Computer technology was first introduced to the Hong Kong commercial sector about 1965. Computers in the Hong Kong education sector, particularly at the Secondary School level, came even later. It was in 1980 that the Hong Kong Government announced plans to introduce computer studies in Secondary Schools. The Computer Studies Pilot Scheme was officially launched in 1982 when thirty government-aided schools were provided with a batch of microcomputers. Since then, more and more schools were equipped with microcomputers each year until now when there are more than two-thirds of the government aided Secondary Schools that have computer systems; of these schools, more than half have their local area network. At the tertiary level, although their computing facilities started much earlier and the institutions were better equipped than the Secondary Schools, the use of computers is mostly restricted to being a subject of its own, as a fast computing instrument and as a data and word processing machine. Computer Assisted Learning (CAL) across disciplines in school is not yet fully recognized. The author, with his interest in, and enthusiasm for, computers, saw the invaluable contribution that the use of microcomputers in education could make and then pursued his research in this field.
1.2 Author's Background

The author, being brought up and having had his Primary and Secondary School education in Hong Kong, pursued his first university degree in Applied Mathematics in the United States of America starting in 1973. Holding his Bachelor of Science from the University of Wisconsin in 1976, he furthered his study as a graduate student in the University of Missouri. During his two-year Master's program, he was offered by the Department of Mathematics a position as Graduate Teaching Assistant from which substantial teaching experience was gained. When he graduated, he returned to Hong Kong and became one of the teaching members of the Hong Kong Polytechnic in the Department of Mathematical Studies and is so still. In the Polytechnic, the author mainly teaches Engineering Mathematics to the early years, and, due to his speciality in the application of computers in mathematics, he is involved in the teaching of Numerical Analysis and Computational Mathematics at the Higher Diploma and Degree levels. During his teaching career, he completed the two-year part-time in-service course and was awarded the Polytechnic Inservice Teacher Training Certificate.

One of his earliest involvements in the education sector was in December 1978 and during the summer of 1979. The Hong Kong Education Department was planning to introduce Computer Studies in Secondary Schools and offered a series of in-service computer courses for Secondary School mathematics
teachers. The author, together with two other staff members in the Department, was asked to be responsible for the teaching of the courses. An evaluation was later conducted, most participants expressed that they enjoyed the courses very much and found them very useful. It was considered to be a very great success in the promotion of Computer Literacy in the Hong Kong Education sector. By that time, Professor Bajpai was the external moderator of the Department of Mathematical Studies of the Hong Kong Polytechnic; with much encouragement, support and advice from Professor Bajpai, a study of the use of microcomputers in mathematics in Hong Kong higher education was made.

1.3 The Scope of the Thesis

In Chapter 2, the history of the development of computer technology is briefly introduced. It starts from the ancient abacus in China, continues through Napier's logarithm system and Pascal's first mechanical calculator to this century's first electronic digital computer, the Atanasoff-Barry Computer (the ABC) and onward. The development of microelectronic technology is growing so rapidly that there have been more discoveries within these few decades than in the past two thousand years. The Altair, the first personal computer, is the bench mark of today's microcomputer. The use of the computer in the Hong Kong commercial sector started in about 1965. The
Universities and Polytechnics commenced a little bit later and most Secondary Schools only had their first computer in the beginning of the 1980s. In 1985, the Chinese University of Hong Kong opened its first microcomputer laboratory which provides computing facilities to students similar to the library from which students borrow printed material.

The progress of computer literacy in Hong Kong is introduced in Chapter 3. The Computer Studies Scheme at Form 4 (Grade 10) level is a two-year computer subject program on its own and was promoted by the Hong Kong Education Department in 1982. In the first few years, the response was encouraging but, in recent years, it was noticeable that some students are starting to refrain from taking this subject due to lack of continuation at the Form 6 level. The Government sees this problem and is reviewing the possibility of introducing a further study of this subject at the Form 6 level. A study of the implementation of computer literacy in the lower Forms is also in progress, the emphasis is planned to let the students use the computer as a tool. At the tertiary level, Computer Studies and Information Technology courses prepare students with updated knowledge in computer technology and applications to meet their needs in their future careers. Future expectations and planning are also considered.

In Chapter 4, the results of a survey conducted by the author are given. The survey is on the computer awareness
of the Hong Kong Polytechnic students. The sample involved 336 Year 1 students in five different academic departments and spanned 4 years. Some trends in the students' awareness of computers are obtained. Also, this survey gives the author the idea of what is the most suitable subject in mathematics to be used in producing Computer Assisted Learning (CAL) packages.

The development and the approach of the first CAL package is explained in Chapter 5 of the thesis. The educational theory in the effectiveness of CAL is established. The aims and objectives of the package are laid. Since the last survey reveals that many students tend to believe that computers have the advantage of performing very accurate computations, the content of the first CAL lesson considered how errors would occur when applying the computers to Numerical Analysis techniques. Different types of error, such as rounding error and truncation error, are explained. Various considerations in the development of the software, such as the computer system and language, are discussed. According to behaviourists, interaction with the computer would involve the student and thus hold his or her interest longer, while the feedback made possible by the computer would facilitate learning, therefore hands-on examples are embodied in the lesson.

Chapter 6 gives the detail evaluation of the students after they have been using the package. Although the evaluation
includes conversations with the students and direct observations from students using the package, detailed analysis of data is possible only on quantitative answers from the questionnaire. Feedbacks and subjective opinion on using computers in mathematical education and on the suitability of applying CAL in the Hong Kong education environment are discussed.

Various aspects of Computer Assisted Learning are investigated in Chapter 7. The essence of the two milestones in CAL, the PLATO and MIME projects, are studied, and the philosophy of CEL (Computer Enhanced Learning), as termed by Professor Bajpai, is also elaborated. Furthermore, different types of CAL, their advantages and disadvantages are explored.

Chapter 8 explains the second CAL package to be used by the Hong Kong Polytechnic students. The topic on numerical integration in Numerical Analysis is chosen. Numerical Analysis is defined as being to devise and evaluate numerical techniques and, in particular,

"... to study their convergence and error."

But it is the author’s experience that students tend to believe that results obtained by the computer are absolute. Therefore, the second piece of software details the error terms of two algorithms in numerical integration when using
computing machines. Graphic illustration is the theme of this package. Student's hands-on verifications are encouraged and are provided in appropriate places in the lesson.

The evaluation of results and achievements after using the second package is described in Chapter 9. Quantitative results are analyzed using the SAS (Statistical Analysis System) package. Students' comments and suggestions on the open-answer questions are presented.

The last chapter, Chapter 10, gives the conclusion of this research. A comparison of the two evaluations in terms of similarity and diversification is highlighted. The effectiveness and considerations in all aspects of the use of microcomputers in mathematics in Hong Kong higher education is summarized. Recommendations and suggestions for further research are deliberated.
2.1 Yesterday, Today and Tomorrow

Today we tend to think of the computer as an electronic device, but the earliest development of computers was based on mechanical or electromechanical technology. People began to consider the use of electronic components in computing machines only after these devices reached the stage where reliable parts started to become available.

Humans had turned to mechanical aids to overcome the difficulty of performing calculations more than two thousand years ago -- the abacus was used in China as early as 500 BC. Ever since, man has been exploring different methods for manipulating the cumbersome number system. In the early seventeenth-century, a multiplying aid, known as Napier's Rods, or Napier's Bones was developed. Based on John Napier's discovery of the logarithm system, the first known linear slide rule was produced in 1654. This invention had taken a giant step forward in improving the speed of computing. Although the first successful mechanical calculator was constructed by Blaise Pascal and was
completed in 1642, the primary conception which this machine introduced was only the mechanization of the carry-over that is needed in addition and subtraction. A direct multiplication machine was not made until 1671, by Leibniz.

The construction of the electronic digital computer was led by the development of the digital electronic machine, known as the Atanasoff-Barry Computer (the ABC), by John Vincent Atanasoff and Clifford Barry. Although the ABC, completed in the spring of 1942, comprised an arithmetic unit and was constructed from more than 200 vacuum tubes, it was capable of only performing addition and subtraction. It differed from a calculating machine only by the card reader and punch next to the console and the two limited drum memory units.

Although there were several "large-scale" electromechanical computer machines designed and operational during that time, the ENIAC, the Electronic Numerical Integrator and Computer, and the people involved in its design and construction were the direct cause of a huge advance in the development and use of electronic computing devices. The ENIAC, considered to be the most complex single piece of equipment at that time, was started in 1943, completed in 1945 and formally dedicated in February 1946. This huge monster took about 18,000 radio tubes and 1,500 electrical relays, together with other related equipment such as plugboard and wiring, and that occupied three sides of a room approximately thirteen by six metres.
Since this big stride, the major differentiation between a computer and a calculator is that the computer can perform long sequences of computations without human intervention and can make certain logical decisions that can even alter its future actions.

After the initial invention of the electronic digital computers, the development of this technology grew so rapidly that it seemed to get out of control. New ideas and new terms, such as batch processing, multiprocessing, timesharing and operating system came one after the other. As more people used the computer, more requirements were met, more breakthroughs were made -- larger memory, smaller size and speed. However, there was an annoying problem in the computer era in the 1950s and 1960s, and that was the inaccessibility of computers. The cost of this new born starlet was such that only the largest institutions -- universities, corporations, research institutions and government agencies -- could afford it.

But all this was changed in 1963 when the Digital Equipment Corporation (DEC) introduced the PDP-8, or the Programmed Data Processor Model 8, the first successful minicomputer. It was a very limited device compared to a mainframe computer, but it cost only a fraction of the price.

Some thirteen years later, the first personal computer, the MIT (Micro Instrumentation and Telemetry Systems) Altair,
was introduced. Despite the fact that the first Altair only had 256-byte basic memory, and that only programs in machine code were allowed, the invention marks the history of the microcomputer and is continuing to reshape the computing world.

Today, a computer with thousands of times increase in memory capacity and performance compared with that which occupied a complete room thirty years ago can now be put on a man’s lap. The speed of a microprocessor is in terms of millions of instructions per second. Nobody knows what tomorrow’s computer world will be. A computer that can speak and communicate with our daily language is not far off.

2.2 Hong Kong in General

Computer technology was introduced to the Hong Kong commercial sector about 1965 when the Hong Kong and Shanghai Bank Corporation first computerized its savings account department. People simply started to enjoy the convenience of being able to deposit and withdraw from their passbook accounts in any of the bank’s branches. Hong Kong and Shanghai Bank’s bold move changed the way in which banks conducted their business. Other banks and corporations followed suit and the computer started to gain popularity and widespread use, making the need for computing equipment multi-fold.
The Hong Kong microcomputer market started to flourish in the late 1970s, synchronizing with the pace of development of microcomputer applications in other parts of the region. In the early days, the microcomputer market in Hong Kong was overwhelmed by home computer users or hobbyists, with most of the machines sold being used for playing video games. From 1982, when IBM announced its first 16-bit personal computer, business applications began to emerge as a more important market sector for microcomputer systems. Just looking at the figures -- starting in 1982, business personal computers have been delivered to Hong Kong at the rate of 8,000 to 10,000 units per year for three successive years, yielding an accumulated installation base of about 35,000 units by 1985 [1] -- gives a rough picture that Hong Kong is desperate in computerization.

Today, all the banks, corporations and even some small firms in Hong Kong are using computers to run their businesses. Office automation is a must.

2.3 The Secondary Schools

Following the recommendation of the White Paper on the Development of Senior Secondary And Tertiary Education (Hong Kong) 1978 [4] that more technological subjects should be put into the secondary school curriculum, the Hong Kong Education Department started the implementation of the Phase
One Computer Studies Pilot Scheme at Form 4 (Grade 10) level in 1982. One of the aims of this scheme is to provide computing facilities for secondary schools upon the recognition of the importance of introducing this new technological subject into the lower level curriculum.

2.3.1 The Computer Studies Pilot Scheme

In the initial stage of the Computer Studies Pilot Scheme, the Phase One stage carried out in August 1982, a total of thirty secondary schools was selected. Each of the schools was provided by the Education Department with nine sets of Atari 800, ten disk drives, two printers and nine household television receivers. A course with class size of forty students was started in September that year using this very limited computer equipment. The objectives and syllabus of the course, which spanned two academic years, has been adopted by the Education Department and will be discussed in the next chapter.

Phase Two of the Pilot Scheme was launched in the academic year 1984-1985. This time, another 75 schools were selected, and 11 sets of APPLE IIe computers, 12 disk drives, 2 printers and 11 composite video colour monitors were provided. In order to provide two more sets of microcomputers to the pilot scheme schools using Atari
computers to make it up to 11, five of the 30 schools were chosen by lot to replace their computers by the new ones from the APPLE Computer, Inc., so as to distribute these Atari computers to the other 25 schools, making a total of 80 schools using APPLE IIe computers and 105 schools offering Computer Studies.

In 1985-86, the scheme was again extended to another 105 schools. Each school was provided with 11 sets of BBC model 8 computers with second processor, 12 disk drives and 11 RGB monitors. The computers were linked together by a network with files and printer servers. After that, the Education Department invited those schools not yet offering Computer Studies to join the scheme. Fifty-five schools were approved to offer this course later. The budget used to carry out this scheme so far has been about HK$30 million.

2.3.2 Exceptional Cases

Not all the schools wanted to be included in the Computer Studies Pilot Scheme. Some schools did not want to join due to school policy, or to the lack of experienced teachers in this area, or to insufficient room to accommodate the computer equipment. Other schools, such as La Salle Secondary School, which had an IBM mainframe computer around 1980, and the St. Paul's Secondary School, which had introduced computer education in 1980, had made a good two
year start ahead of the Computer Studies Pilot Scheme pushed by the Hong Kong Government.

In the following case, a brief history of the development of the Computer Department of a secondary school, the St. Paul’s College, which was also not in the scheme, is described.

<table>
<thead>
<tr>
<th>Date</th>
<th>Hardware Development</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nov., 81</td>
<td>The school’s first computer -- an APPLE II microcomputer with one disk drive and a printer.</td>
</tr>
<tr>
<td>Sept., 82</td>
<td>Four more sets of computer were bought -- all Radio Shack TRS-80 model III. Three of these had single disk drives and the fourth one had a dual disk drive. An additional dot matrix printer was installed.</td>
</tr>
<tr>
<td>Jan., 83</td>
<td>New computer room was completed.</td>
</tr>
<tr>
<td>Sept., 83</td>
<td>Six new computers were bought -- all BBC microcomputers model B, connected in a network.</td>
</tr>
</tbody>
</table>
| Nov., 83  | A wide carriage printer and five more BBC microcomputers were bought. }
In order that students could have more access to the computers, two BBC microcomputers with cassette recorders and one Radio Shack microcomputer were moved to the library.

In St. Paul's College, computer classes were all after normal school hours in order not to upset the already tight timetable schedule. Despite that, the students still had an average of about one and a half hours of practical time in the computer room each week. The opening hours of the computer room for individual students on normal school days were:

<table>
<thead>
<tr>
<th>Day</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon., Tue. Fri.</td>
<td>3.40 - 5.00 p.m.</td>
</tr>
<tr>
<td>Thur.</td>
<td>3.40 - 4.30 p.m.</td>
</tr>
<tr>
<td>Sat.</td>
<td>9.00 - 12.00 noon(*)</td>
</tr>
<tr>
<td></td>
<td>2.00 - 3.30 p.m.</td>
</tr>
</tbody>
</table>

(*) Two one-and-a-half-hour sessions for Computer Studies Class students only.

They managed to operate the computer room very successfully with the help of some rules for the students to observe. These rules included: advanced booking needed, unauthorized use prohibited, and no show penalty enforced, etc.

Details concerning the history of development of the
computer department, and the opening hours and rules for the computer room in St. Paul's college are included in the Appendix A.

2.4 The Polytechnics in Hong Kong

There are two Polytechnics in Hong Kong. The Hong Kong Polytechnic, a former Hung Hom Technical College of Hong Kong, was established in 1976. The second one is the City Polytechnic of Hong Kong, which had its first batch of students in 1984. Their computing facilities will be discussed separately in the following subsections.

2.4.1 The Hong Kong Polytechnic

Due to the historical background of the Hong Kong Polytechnic (HKP), it already had limited computing facilities in its establishment. As demand increased in the HKP as well as in the two universities in Hong Kong, the Hong Kong Government decided to establish a Universities and Polytechnic Computer Centre Limited, (UPCC) and to purchase a mainframe computer, an ICL 1900s, for the use of the three institutes in order to optimize resources. Although the UPCC was sited within the HKP campus, sometimes the turn around time for even small computer jobs took overnight. As the HKP believed that its requirements comprised mainly a
huge amount of small jobs from novices, some kind of
time-sharing on-line system would be more suitable.
Therefore, it bases its requirements on minicomputers; this
policy has not been changed ever since.

By the late 1970s, the computer centre of the HKP stabilized
its main user support on three DEC PDP 11/70 minicomputers.
For security and efficiency reasons, the three computers did
not link together directly. There was one dedicated to
business application, one for scientific application and the
third one for administration use.

In the early 1980s, the three PDP computers could not
satisfy the ever increasing computing demand and they had
also reached the age of retirement. A DEC VAX 11/750 was
then in service. Later, more VAX minicomputers were in use,
with some individual departments, such as the Department of
Mathematical Studies, even having their own VAX system.
Currently, other than the departmental owned computers, the
Computer Centre of HKP is supporting two VAX 11/750, one VAX
11/780, one VAXCluster with three processors and a DG
MV10000 for general use. There are also other systems
dedicated to special applications, such as library indexing,
student records, which are not of interest here.

There are also quite a number of microcomputers spread
around the campus. Similarly to many other institutes in
the world, individual departments may have their own computer laboratory accommodating all different models for their specific purposes. However, the HKP Computer Centre provides both software and hardware support to some commonly used computer models.

2.4.2 The City Polytechnic

The City Polytechnic of Hong Kong (CPHK) was founded in 1984, as was its computer centre which is a centralized facility and support division. In that year, four million dollars of equipment was installed. These included a Perkin Elmer 3210 system, a NCR Tower XP and a Data General MV4000 system. There were about 80 terminals. In the second year, another NCR Tower was bought and the Data General MV4000 was upgraded to MV10000 which has four times the capacity of the smaller model. A DEC VAX 785, which is the backbone for CPHK's CAD/CAM (Computer Aided Design/Computer Aided Manufacturing), was installed. Microcomputers also got their shares. Ninety IBM-PC compatibles for the microcomputer laboratory, library and departments, were bought. These added up to another HK$5 million in 1985. As CPHK currently is located temporarily in a commercial building, the new campus, expected to be operational in 1988, will have a fully networked computer centre occupying 4,000 square metres of floor space.
2.5 The Universities

There are two universities in Hong Kong, the Hong Kong University (HKU) and the Chinese University of Hong Kong (CUHK). As has been mentioned before, the universities and the HKP were sharing the same mainframe computer under UPCC in the early days. However, the computer centre of CUHK felt the increasing demand of computing facilities, and had their IBM 3031 by 1978. Later, the HKU also bought their UNIVAC 1100. In order to enhance computing facilities among the three institutes, a networking system providing cross-institute batch job submission among the mainframes was completed. The ICL 1900s was no longer in service and the existence of UPCC was only by name.

A few years ago, on October 25, 1985, the CUHK officially pronounced the opening of its United College (United College is one of the three constituent colleges in the Chinese University of Hong Kong) Microcomputer Laboratory (UCMCL) to promote computer literacy. The equipment of UCMCL includes:

18 IBM PC's, each with 256KB memory, 2 floppy disk drives, 1 Everex Graphics Edge colour card; many of the PC's with colour screens and 8087 arithmetic chips.

2 IBM AT's, each with 512KB memory, 1 360KB floppy
disk drive, 1 1.2MB floppy disk drive, 10 MB hard disk, colour screen, Everex Graphics Edge colour card, 80287 arithmetic chip.

2 Toshiba 1100 portable personal computers with 512KB memory and 5.25-inch disk drive attachment.

2 APPLE Macintoshes each with 512KB memory, 1 floppy disk drive, monochrome screen and Mouse.

3 Toshiba P1351 printers, 3 Epson FX80+ printers, and 1 APPLE Imagewriter.

The laboratory is open from 9.00 a.m. to 9.30 p.m. Mondays to Fridays and 9.00 a.m. to 12.30 p.m. on Saturdays. It functions in a way such that students come to borrow machine time just as they borrow books from the library.
CHAPTER 3

Computer Literacy In Hong Kong Education Sector

3.1 The Secondary Schools

The introduction of computer literacy by the Hong Kong Education Department followed the recommendation of the White Paper on the Development of Senior Secondary and Tertiary Education (Hong Kong) 1978 [4]. A phase one Computer Studies Pilot Scheme at Form 4 (Grade 10) level was implemented in 1982. But prior to this Computer Studies Pilot Scheme being introduced to the secondary schools, a series of in-service training courses in computer studies for school teachers had been jointly organized by the Education Department, the Computer Centre of CUHK and the Department of Mathematical Studies in HKP in 1978 and 1979. The author was one of the lecturers in the courses at that time. In addition to the training courses, a one-year pre-pilot course had been provided for students in the St. Paul’s Secondary School. Experience gained was taken into consideration for the implementation of the Computer Studies Pilot Scheme. The objectives and syllabus will be discussed in the next subsection.
3.1.1 The Objectives and Syllabus

The objectives and syllabus of the Computer Studies course were jointly set out by teachers and educators from tertiary and secondary institutions and representatives from the commercial sectors. They are revised from time to time so as to keep pace with the rapidly changing world of computer technology.

The objectives of the course are:

1. To provide opportunity for the study of modern methods of information processing so that students may understand and apply this rapidly growing technology.

2. To acquaint students with the uses and limitations of computers.

3. To develop among students problem solving skills through interaction with computers.

4. To encourage an understanding of the implications of computers in the modern world.

5. To prepare students who wish to go on to further studies in computer science.
It should be noted that this course is largely independent of any other subject in the secondary school level curriculum. In particular, it contains very little mathematics. The content of the syllabus can be classified into three categories: namely general knowledge, programming and computer application. They include:

1. Evolution of information processing
2. Introductory computer concepts
3. Flowcharting
4. Programming in BASIC
5. Input/Output, coding and storage of information
6. Data processing
7. Computers in the modern world

The detailed syllabus can be found in the Appendix B. However, from the outline of the contents, it can be seen that it is very demanding. It covers a rather broad spectrum of computer knowledge. The recommended time allocation is four periods per week. The syllabus spans two years (Form 4 and Form 5), and so taking an average school year as 27 teaching weeks, a total of 216 periods, including hands-on program development and testing, are available. It is quite challenging to both the students and the teachers.
3.1.2 The Extra-Curriculum

The microcomputer activities in the secondary schools are not confined to teaching. A group of teachers and educators from tertiary and secondary institutions formed The Hong Kong Association for Computer Education in 1981. Anyone with an interest in computers is welcome to join. Members are encouraged to come to the monthly meeting to share their experiences and to exchange ideas among themselves. The welfare section also takes care of group purchase of computer accessories such as diskettes, papers and coding forms.

One of the biggest events for Computer Studies teachers is the yearly conference on "Microcomputers in Education" organized by the Hong Kong Professional Teachers' Union. The conference includes plenary talks on selected topics, discussions and reviews on various areas, demonstration of computer software especially designed for educational purpose, and last, but not least, an exhibition of hardware products and the winning entries of a creativity contest for students in computer applications.

Computerization of school administration using the existing microcomputer facilities is also in progress. There are software packages especially designed for Hong Kong secondary schools, such as a Students' Data Management
System, and a Library Management System which can handle both Chinese and English titles. Some packages are developed by the private sector and software houses and written more professionally, but their subscription fee is substantial. Some packages are written by teachers, but owing to their limited programming experience, these packages are not very sophisticated. But usually they are tailor made and easy to obtain. A prototype can be given by the Sung Tsun Middle School in Sai Kung, New Territories, who have developed a Mark System to calculate student marks and keep student records since 1983. After thorough testing and implementation, they now make it available at a nominal charge to any school that would like to use it. A copy of their package introduction and sample outputs are included in the Appendix B.

There is another package worth-mentioning written by amateur programmers, but professional teachers, called Automation Multiple Choice Question Selection System (Author’s translation from Chinese). This system not only can select multiple choice questions from a question pool and give the model answers, it can also update the difficulty level of each question as well as identify which question has appeared in what test to ensure intelligent selection. More questions on various subjects and levels are now called for to be included in the question bank and are available to all subscription school members.
3.1.3 Feedbacks

Computer Studies has been offered to Form 4 and Form 5 students for more than four years. So far, three batches of students have already taken the Hong Kong Certificate Entrance Examination (more or less equivalent to the standard of that of the G.C.E. O-level) and the results have proved to be most encouraging. They have performed more than satisfactorily. However some teachers have observed that some very bright students are starting to refrain from taking this subject due to the fact that there is lack of continuation at the Form 6 level. They do not want to invest their time in a subject that will not be offered in Form 6 and hence not help them to gain entry to the universities in Hong Kong. Some schools decline to enter into the Computer Study Scheme for this same reason. The CUHK has a seminar on "Computer Studies at the Form Six Level" to discuss this matter. The Government sees this problem and the Examination Authorities have invited opinions from secondary school principals on the introduction of the subject in the Form 6 level. The answer is in the affirmative on the condition that resources are obtained from the Government.

The resources to which the school principals are referring are not only budgeting, but also with the availability of computer teachers. The commercial world is in urgent need of computer people. Clearly not too many computer major
graduates will be interested in the teaching profession. If Computer Studies are offered at the Form 6 level, it will increase the pressure of the already tense crisis.

At present, there are regular computer in-service training courses each year and also seminars on Data Processing and Data Structure and Algorithm for secondary school teachers. But it is still hoped that the Education Department can look ahead to start up long term plans, such as a Teacher Release Scheme or a Teacher Development Program, as in other countries [6], for computer studies teachers in order to provide training, refreshing and updating to the rapidly changing computer technology.

3.1.4 Further Expectations

As it has been observed, the three batches of computer hardware provided by the Education Department to the schools are all different. The software run in one machine may not be run in another. The problems of incompatibility and importability make the exchange of packages and programs impossible, hence unnecessary duplication of effort is unavoidable. But there is the argument (which is also true) that the best suited computers were chosen at each time and that it was difficult, if not impossible, to foresee what the trend of microcomputer technology would be in two or three years' time. However, it would still be worthwhile to
make the effort to ensure that software packages would not be stalled due to hardware incompatibility. A standardization may benefit future development, such as ultra-school networking and communication with the universities in Hong Kong or even internationally.

The current Computer Studies syllabus for Form 4 and Form 5 is already very demanding. Therefore if a syllabus for Form 6 is to be set, it should consist of various options from which the students can choose. It should emphasize creativity, logical thinking and intelligent application rather than be encyclopaedic. If the Education Department is working on the implementation of Computer Literacy at the levels of Form 1 to Form 3, they should let the students use the computer as a tool, to dispel fear of the computer world, to fascinate them with modern technology and to generate curiosity. Nobody wants to create computer phobia out of computer literacy.

3.2 The Polytechnics

In the Hong Kong Polytechnic, the Department of Computing Studies is responsible for the servicing of all computing subjects to all other departments. Almost all the students are required to take at least one computing subject, some may need more. Take the Higher Diploma students in the Department of Mathematical Studies as an example. In the
first year, the students are required to take three computing courses, which include the fundamental concepts and a computer language, PASCAL, and Microcomputer and Applications. These three courses should be able to equip them with the general programming technique, and enough computer background for later studies and applications. In the second year, there are two more compulsory computer subjects, "Introduction to Information Processing" and "Introduction to System Software" which include data structure, file handling techniques and the operating system concepts. By this time, the students should have enough computing knowledge to use what they have learned in other subjects, such as Numerical Analysis, Operations Research and Statistics. In the third year, there are two other courses offered, but they are optional. One concerns data communication and networking, while the other is the project. Almost all the students take these two courses irrespective of the fact that they are not compulsory. The syllabi are updated constantly and a revision has just been made. The author is one of the two members of staff in the department to be responsible for revising this curriculum. Detail syllabi are included in the Appendix B.

In the summer of 1985, The Department of Computing Studies had an "Information Technology Summer Course" for HKP staff. The course offered the following topics: BASIC, FORTRAN, DATABASE, WORDPROCESSING and SPREADSHEET. Staff could take one or more topics depending on how much time they had. The
course included a series of films showing independent studies and working in the microcomputer centre. As found by talking with staff who had taken the course and observations by the author, it did not run very successfully. Enthusiasm died away gradually due to lack of consultation when problems arose, no aims and no incentive due to continuous frustration.

In the City Polytechnic of Hong Kong, the Computer Studies Department is one of eight academic departments. However, they have placed great emphasis on computer literacy. Due to their short history in the Hong Kong community, it seems too early to give any conclusion on their blooming achievements.

3.3 The Universities

The two universities in Hong Kong did not introduce computer studies as a major subject until 1975. They were at least ten to fifteen years behind other places in this respect. The University of Hong Kong offers a Bachelor of Science (Computer Studies) degree. Most graduates from this course are employed in the commercial field. In order to cater for those who want to take up their careers in the teaching profession, the Faculty of Science in HKU has implemented a new B.Sc curriculum putting more stress on microcomputer applications and related subjects, hoping that students will
be able to enhance their ability to use the microcomputer on computer studies education. The first batch of students on this course commenced their studies in September 1987.

In the Chinese University of Hong Kong, due to the establishment of the microcomputer laboratory UCMCL (see Chapter 2), students in Computer Science and Electronics, as well as other faculties, have in addition access to the computer world. They can have hands-on experience of many popular commercial packages such as DBase III, Symphony, Lotus 1-2-3, etc.

3.4 Follow-up Training

Many employers find that employing fresh graduates often poses a problem due to their insufficient practical experience in real life electronic data processing (EDP) projects. In view of this, the Vocational Training Council is setting up an EDP Training Centre for people who are just starting their careers in this field. The main emphasis will be working on close-to-real-situation projects which include planning, discussions and presentation. The EDP Training Centre also plans to run evening courses for those who have been in the EDP profession for some time but have not had the opportunity to obtain tertiary education. These people will then be able to take up more responsible positions on completion of the course.
Survey On Computer Awareness Of HKP Students

4.1 Why the Survey

It is now more than forty years since the innovation of the first computer. It was in 1976 that the first microcomputer was introduced in the United States of America. Therefore, for some time now people have had some idea as to what a computer is.

The author started this survey in 1983, just a year after the Computer Studies Pilot Scheme was implemented. Consequently there should be no one involved in the survey who had benefited from the scheme. The same survey was conducted in successive years until 1986. During these years, the number of students having formal training in computer studies should have increased. It was in the author's interest to see if there were any significant changes in computer awareness in Hong Kong Polytechnic students during the period just before and after Computer Studies was introduced in the secondary school. The result of the survey, however, should not be considered as an indication of computer awareness in all Hong Kong students
because it is somewhat biassed in the sense that the students in this survey were only those already admitted by Hong Kong Polytechnic; others with various educational backgrounds had not been included.

Based on the results of the survey, the author would also be able to decide later how the Computer Assisted Learning packages were going to be written. What topics should go in? Should detailed instructions of the operation of the machine be included? How much computer knowledge on the part of the students can be assumed? The findings of the survey will now be discussed.

4.2 The Surveys

The survey covered four academic years, 1983/84, 1984/85, 1985/86, 1986/87, and was given in the form of answering written questions. A questionnaire was designed and used in the autumn of 1983. This was referred to as Questionnaire One, as there were other questionnaires connected with the feedbacks of students using the Computer Assisted Learning packages. The other questionnaires will be described in subsequent chapters. A copy of Questionnaire One: 'Computer Awareness of Hong Kong Polytechnic Students' is attached in the Appendix C.

The Questionnaire comprised nine questions occupying two
sheets of papers. The first page had five multiple choice questions, and the second page contained four open-answer questions. The following questions were asked:

1. Sex
2. Age group
3. How much do you think you know about computers?
4. Have you ever used a computer or an item that is controlled by computer?
5. What advantage(s) do you think a computer has?
6. An ordinary person has I.Q. around 100. What I.Q. score would you give to a computer?
7. What things would you like to be processed by computer?
8. Name 5 computer languages, spell them in full
9. Name 5 computer manufacturers in full names, give 1 computer model for each manufacturer.

Details of each question are going to be discussed.

Question 1 was to classify the student either male or female. Question 2 was to find the distribution of their ages. Since they were expected to be Polytechnic students, only 4 groups: below 20, 20-21, 22-23, 24 or above, were classified. These two questions would provide the statistics on the sample students.

Question 3 gave a six-point scale, ranging from 'nothing' to
'very well', to indicate how much the students thought they knew about computers. Question 4 was simply a 'Yes' or 'No' question on whether they had used a computer or an item that was controlled by a computer.

Question 5 gave a listing of choices that they would think of as the advantages of a computer. These choices were

- Fast in speed
- Accurate in calculation
- Can solve problems for you
- Easy to use
- No need to take care of

Questions 6 to 9 are open-answer questions. The students could fill in their own answers.

These nine questions could easily be put into four categories according to their nature. They were: Personal Particulars, Questions 1 and 2; Self-judgement, Questions 3 and 4; Computer Ideas, Questions 5, 8 and 9; and Their Computer World, Questions 6 and 7.

4.3 The Target Population and the Sample

The target population for this survey was all the year one students in Hong Kong Polytechnic who would necessarily
have to use the computer one way or the other during their courses. However, due to limited resources, it was impossible for the author to include the entire population. But it was the author's intention to include as many different disciplines of students as possible. The survey commenced in October 1983 and continued until October 1986. The sample involved Departments of Applied Science, Business and Management, Electrical Engineering, Mathematical Studies, and Mechanical Engineering, and the students involved were following full-time higher diploma, part-time evening endorsement and part-time degree programs. The questionnaires were distributed in person by the author during the first available period at the beginning of a term. The students were allowed a maximum of fifteen minutes to complete the questions. Discussion of answers among students was discouraged. Since question papers were collected immediately after they had been completed, zero non-response rate was assumed. However, some questions in the questionnaires may have been found to be unanswered, these would be treated separately. A total of nine classes and 336 respondents were obtained. Table 4.1 shows by year the departments of these 9 classes taking the survey together with their programs.

This sample of nine classes, which spread through four academic years from 1983/84 to 1986/87, had actually come from five different departments following five different programs. That means the same program may have been
### Table 4.1 Departments In Chronicle Order

<table>
<thead>
<tr>
<th>Class No.</th>
<th>Year</th>
<th>Department</th>
<th>Mode of Program(*)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1983/84</td>
<td>Business &amp; Management Studies</td>
<td>FT HD</td>
</tr>
<tr>
<td>2</td>
<td>1983/84</td>
<td>Mathematical Studies</td>
<td>FT HD</td>
</tr>
<tr>
<td>3</td>
<td>1983/84</td>
<td>Mechanical Engineering</td>
<td>PTE END</td>
</tr>
<tr>
<td>4</td>
<td>1984/85</td>
<td>Mathematical Studies</td>
<td>FT HD</td>
</tr>
<tr>
<td>5</td>
<td>1985/86</td>
<td>Electrical Engineering</td>
<td>FT HD</td>
</tr>
<tr>
<td>6</td>
<td>1985/86</td>
<td>Mechanical Engineering</td>
<td>PTE END</td>
</tr>
<tr>
<td>7</td>
<td>1985/86</td>
<td>Applied Science</td>
<td>PT DEG</td>
</tr>
<tr>
<td>8</td>
<td>1986/87</td>
<td>Applied Science</td>
<td>PT DEG</td>
</tr>
<tr>
<td>9</td>
<td>1986/87</td>
<td>Mathematical Studies</td>
<td>FT HD</td>
</tr>
</tbody>
</table>

(*) FT HD - Full-time Higher Diploma  
PTE END - Part-time-evening Endorsement  
PT DEG - Part-time Degree

### Table 4.2 Distribution Of Respondents By Class

<table>
<thead>
<tr>
<th>Class No.</th>
<th>Number of Respondents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>32</td>
</tr>
<tr>
<td>2</td>
<td>29</td>
</tr>
<tr>
<td>3</td>
<td>41</td>
</tr>
<tr>
<td>4</td>
<td>43</td>
</tr>
<tr>
<td>5</td>
<td>38</td>
</tr>
<tr>
<td>6</td>
<td>29</td>
</tr>
<tr>
<td>7</td>
<td>22</td>
</tr>
<tr>
<td>8</td>
<td>26</td>
</tr>
<tr>
<td>9</td>
<td>76</td>
</tr>
</tbody>
</table>

**Total 336**
Table 4.3  Distribution By Department

<table>
<thead>
<tr>
<th>Department</th>
<th>Number of Respondents</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Applied Science</td>
<td>48</td>
</tr>
<tr>
<td>B Business &amp; Management</td>
<td>32</td>
</tr>
<tr>
<td>C Electrical Engineering</td>
<td>38</td>
</tr>
<tr>
<td>D Mathematical Studies</td>
<td>148</td>
</tr>
<tr>
<td>E Mechanical Engineering</td>
<td>70</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>336</strong></td>
</tr>
</tbody>
</table>

included in the survey in more than one year. For instance, the students in Department of Mathematical studies were included in 1983, 1984 and 1986; Mechanical Engineering in 1983 and 1985; Applied Science in 1985 and 1986. In so doing, the author would also be able to make comparisons with students in different years but in the same program and department. The numbers of respondents of each of the nine classes is shown in Table 4.2, and the number of respondents of each of the five departments is shown in table 4.3.

Please note that in these tables, the nine classes and five departments are numbered and labelled respectively for simplicity and will be used for future identification and reference.
Among the respondents, it was found that more than three quarters were male students and about one quarter were female students. In particular, almost all the students in the two engineering departments were masculine except one. The other departments also showed domination of one gender over the other. Table 4.4 and Table 4.5 give the distribution of respondents in sex by class and by department respectively. The bracketed values give the percentages. Percentages shown are rounded to the nearest integer and consequently they may not add up exactly to 100.

Tables 4.6 and 4.7 give the age distribution of respondents by class and by department respectively.

Since the age groups were open-ended, the mean age could not be obtained. However, the modal class of each class and each department was identified by an asterisk. Inspecting the overall figures, the modal class was found to be in the age group between 20 and 21. The calculated mode was 20.85, and the age distribution was skew to the right, resulting in a positive skewness.

Further study of the student ages in individual departments revealed that the majority of students in the Departments of Applied Science and Mechanical Engineering were in the older age group; this might be due to the fact that they were part-time students and most of them would have been working for some time. It is also interesting to note that, possibly
### Table 4.4 Distribution Of Sex By Class

<table>
<thead>
<tr>
<th>Class No.</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>19 (59%)</td>
<td>13 (41%)</td>
</tr>
<tr>
<td>2</td>
<td>18 (62%)</td>
<td>11 (38%)</td>
</tr>
<tr>
<td>3</td>
<td>40 (98%)</td>
<td>1 (2%)</td>
</tr>
<tr>
<td>4</td>
<td>30 (70%)</td>
<td>13 (30%)</td>
</tr>
<tr>
<td>5</td>
<td>38 (100%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>6</td>
<td>29 (100%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>7</td>
<td>18 (82%)</td>
<td>4 (18%)</td>
</tr>
<tr>
<td>8</td>
<td>24 (92%)</td>
<td>2 (8%)</td>
</tr>
<tr>
<td>9</td>
<td>41 (53%)</td>
<td>35 (47%)</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>257 (76%)</td>
<td>79 (24%)</td>
</tr>
</tbody>
</table>

### Table 4.5 Distribution Of Sex By Department

<table>
<thead>
<tr>
<th>Department</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>42 (88%)</td>
<td>6 (13%)</td>
</tr>
<tr>
<td>B</td>
<td>19 (59%)</td>
<td>13 (41%)</td>
</tr>
<tr>
<td>C</td>
<td>39 (100%)</td>
<td>0 (0%)</td>
</tr>
<tr>
<td>D</td>
<td>89 (60%)</td>
<td>59 (40%)</td>
</tr>
<tr>
<td>E</td>
<td>69 (99%)</td>
<td>1 (1%)</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>257 (76%)</td>
<td>79 (24%)</td>
</tr>
</tbody>
</table>
Table 4.6  Distribution Of Age-group By Class

<table>
<thead>
<tr>
<th>Class No.</th>
<th>Below 20</th>
<th>20-21</th>
<th>22-23</th>
<th>24 or above</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>22*</td>
<td>13</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>22*</td>
<td>7</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>4</td>
<td>18</td>
<td>19*</td>
</tr>
<tr>
<td>4</td>
<td>19</td>
<td>21*</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>19*</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>13</td>
<td>16*</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>6</td>
<td>5</td>
<td>11*</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>4</td>
<td>6</td>
<td>16*</td>
</tr>
<tr>
<td>9</td>
<td>46*</td>
<td>25</td>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>

Total 99(29%) 108(32%)* 65(19%) 64(19%)

* The modal class

Table 4.7  Distribution Of Age-group By Department

<table>
<thead>
<tr>
<th>Department</th>
<th>Below 20</th>
<th>20-21</th>
<th>22-23</th>
<th>24 or above</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0( 0%)</td>
<td>10(21%)</td>
<td>11(23%)</td>
<td>27(56%)*</td>
</tr>
<tr>
<td>B</td>
<td>6(19%)</td>
<td>22(69%)*</td>
<td>3( 9%)</td>
<td>1( 3%)</td>
</tr>
<tr>
<td>C</td>
<td>6(16%)</td>
<td>19(50%)*</td>
<td>12(32%)</td>
<td>1( 3%)</td>
</tr>
<tr>
<td>D</td>
<td>87(59%)*</td>
<td>53(36%)</td>
<td>8( 5%)</td>
<td>0( 0%)</td>
</tr>
<tr>
<td>E</td>
<td>0( 0%)</td>
<td>4( 6%)</td>
<td>31(44%)</td>
<td>35(50%)*</td>
</tr>
</tbody>
</table>

Total 99(29%) 108(32%)* 65(19%) 64(19%)

* The modal class
due to their working experience, they had a higher score in some questions in the questionnaire than those of the younger groups. Further analysis will be discussed later.

Taking both sex and age together, it was found that there were no female students in the "24 or above" age category and there were only 3 female students in the "22-23" age category. Compared with the males, who had a total of 129 in the two categories, the females represented just over two percent. Therefore it could be concluded that generally speaking, the female respondents were in a younger group.

4.4.1 Tabulation Results on Closed-Answer Questions

In Question 3, it was asked "How much do you think you know about computers?". A six-point scale, 0, 1, 2, 3, 4, 5, corresponding to 'nothing', 'a little', 'a few', 'fair', 'well', and 'very well' was given. The students were asked to choose the entry that they thought was most appropriate to apply to themselves, without comparing with others. It came out that nobody thought that he or she knew about the computer 'very well' and only a few of them (12 students, or about 4% of the total) thought that they knew the computer 'well'. Almost one half of the students (48% of the total) thought that they knew 'a little' (scale point 1) about computers and approximately the same equal percentage of students spread among the 'nothing', 'a few', and 'fair'
categories. The average score was 1.49, or nearly the mid-point of 'a little' and a 'few'. Table 4.8 shows the results of respondents bracketed by the percentage in class order. Table 4.9 gives the results in percentage by year. It was observed from Table 4.9 that the percentage of respondents who thought that they 'knew about computer fair' (scale point 3) increased constantly in three consecutive years. The increase in percentage in 1986 was even higher in the point 4 ('well') category, which was two-thirds of the total. Further analysis will be given in a later section.

In Question 4, students were asked 'Have you ever used a computer or an item that is controlled by computer?'. Irrespective of the fact that there were two non-responses to this question, it was surprising to see that over one-quarter of the students gave the answer 'No'. This might be due to their being unaware that many home appliances nowadays, for example, washing machines, cassette deck players, video recorders, have microprocessors built inside them. And almost all newer models of microwave ovens can be programmed and decide the cooking procedure and cooking time for users. Needless to say, the bank's cash dispenser machines are a kind of computer terminal. Sooner or later, many mechanically operated parts will be replaced by electronics. It has been commented by Chen, T.C. [8] that:
Table 4.8  Distribution On Knowing About Computer By Class

<table>
<thead>
<tr>
<th>Class No.</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14(44%)</td>
<td>13(41%)</td>
<td>3( 7%)</td>
<td>2( 5%)</td>
<td>0( 0%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>2</td>
<td>7(24%)</td>
<td>18(62%)</td>
<td>2( 7%)</td>
<td>2( 7%)</td>
<td>0( 0%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>3</td>
<td>1( 2%)</td>
<td>18(44%)</td>
<td>16(39%)</td>
<td>5(12%)</td>
<td>1( 2%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>4</td>
<td>6(14%)</td>
<td>24(56%)</td>
<td>6(14%)</td>
<td>6(14%)</td>
<td>1( 2%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>5</td>
<td>4(11%)</td>
<td>21(55%)</td>
<td>9(24%)</td>
<td>4(11%)</td>
<td>0( 0%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>6</td>
<td>0( 0%)</td>
<td>12(41%)</td>
<td>7(24%)</td>
<td>9(31%)</td>
<td>1( 3%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>7</td>
<td>0( 0%)</td>
<td>7(32%)</td>
<td>5(23%)</td>
<td>9(41%)</td>
<td>1( 5%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>8</td>
<td>0( 0%)</td>
<td>8(31%)</td>
<td>4(15%)</td>
<td>9(35%)</td>
<td>5(19%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>9</td>
<td>14(18%)</td>
<td>39(51%)</td>
<td>11(14%)</td>
<td>9(12%)</td>
<td>3( 4%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>Total</td>
<td>46(14%)</td>
<td>160(48%)</td>
<td>63(19%)</td>
<td>55(16%)</td>
<td>12( 4%)</td>
<td>0(0%)</td>
</tr>
</tbody>
</table>

Table 4.9  Percentage On Knowing About Computer By Year

<table>
<thead>
<tr>
<th>Year</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>22%</td>
<td>48%</td>
<td>21%</td>
<td>9%</td>
<td>1%</td>
<td>0%</td>
</tr>
<tr>
<td>1984</td>
<td>14%</td>
<td>56%</td>
<td>14%</td>
<td>14%</td>
<td>2%</td>
<td>0%</td>
</tr>
<tr>
<td>1985</td>
<td>4%</td>
<td>55%</td>
<td>19%</td>
<td>20%</td>
<td>2%</td>
<td>0%</td>
</tr>
<tr>
<td>1986</td>
<td>14%</td>
<td>46%</td>
<td>15%</td>
<td>18%</td>
<td>8%</td>
<td>0%</td>
</tr>
</tbody>
</table>
"anything which sells for HK$100 may soon be armed with a computer chip to provide extra (apparent) intelligence, animation, flexibility and sales appeal, if not right in the merchandise, then probably in the container."

Table 4.10 gives the distribution of this question in class order and Table 4.11 gives the percentage of the response grouped in years.

In Question 5, it was asked "What advantage(s) do you think a computer has?" Five choices were given and students could tick one or more options that they thought were the advantages of a computer. These options were:

a. Fast in speed
b. Accurate in calculation
c. Can solve problems for you
d. Easy to use
e. No need to take care of.

Table 4.12 gives the response to this question and Table 4.13 shows the percentage of students' response by year. Since the students might tick one choice, two choices or up to all five choices, therefore the total percentage on all responses may be more than 100. According to the result, over 83% of the students thought that the computer had the advantage of doing things fast, and the percentage decreased
Table 4.10  Distribution Of Question 4 In Class Order

<table>
<thead>
<tr>
<th>Class No.</th>
<th>Yes</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>22(76%)</td>
<td>10(24%)</td>
</tr>
<tr>
<td>2</td>
<td>18(69%)</td>
<td>11(31%)</td>
</tr>
<tr>
<td>3</td>
<td>31(62%)</td>
<td>10(38%)</td>
</tr>
<tr>
<td>4</td>
<td>33(77%)</td>
<td>10(23%)</td>
</tr>
<tr>
<td>5</td>
<td>29(76%)</td>
<td>9(24%)</td>
</tr>
<tr>
<td>6</td>
<td>21(72%)</td>
<td>8(28%)</td>
</tr>
<tr>
<td>7</td>
<td>19(86%)</td>
<td>3(14%)</td>
</tr>
<tr>
<td>8</td>
<td>21(88%)</td>
<td>3(13%)</td>
</tr>
<tr>
<td>9</td>
<td>55(72%)</td>
<td>21(28%)</td>
</tr>
</tbody>
</table>

Total  249(74%)  85(26%)

Table 4.11  Percentage On Question 4 By Year

<table>
<thead>
<tr>
<th>Year</th>
<th>Yes</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>70%</td>
<td>30%</td>
</tr>
<tr>
<td>1984</td>
<td>77%</td>
<td>23%</td>
</tr>
<tr>
<td>1985</td>
<td>78%</td>
<td>22%</td>
</tr>
<tr>
<td>1986</td>
<td>76%</td>
<td>24%</td>
</tr>
</tbody>
</table>
Table 4.12  **Response On Question 5 In Class Order**

<table>
<thead>
<tr>
<th>Class No.</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>24(75%)</td>
<td>17(53%)</td>
<td>9(28%)</td>
<td>4(13%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>2</td>
<td>24(83%)</td>
<td>19(66%)</td>
<td>21(72%)</td>
<td>2(7%)</td>
<td>1(3%)</td>
</tr>
<tr>
<td>3</td>
<td>34(83%)</td>
<td>27(66%)</td>
<td>26(63%)</td>
<td>8(20%)</td>
<td>1(2%)</td>
</tr>
<tr>
<td>4</td>
<td>39(91%)</td>
<td>26(60%)</td>
<td>18(42%)</td>
<td>7(16%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>5</td>
<td>30(79%)</td>
<td>8(21%)</td>
<td>15(39%)</td>
<td>6(16%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>6</td>
<td>24(83%)</td>
<td>12(41%)</td>
<td>14(48%)</td>
<td>5(17%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>7</td>
<td>17(77%)</td>
<td>10(45%)</td>
<td>13(59%)</td>
<td>2(9%)</td>
<td>1(5%)</td>
</tr>
<tr>
<td>8</td>
<td>25(96%)</td>
<td>11(42%)</td>
<td>10(38%)</td>
<td>3(12%)</td>
<td>0(0%)</td>
</tr>
<tr>
<td>9</td>
<td>63(83%)</td>
<td>51(67%)</td>
<td>52(68%)</td>
<td>17(22%)</td>
<td>2(3%)</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>280(83%)</strong></td>
<td><strong>181(54%)</strong></td>
<td><strong>178(53%)</strong></td>
<td><strong>54(16%)</strong></td>
<td><strong>2(1%)</strong></td>
</tr>
</tbody>
</table>

Table 4.13  **Percentage On Question 5 By Year**

<table>
<thead>
<tr>
<th>Year</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>80%</td>
<td>62%</td>
<td>55%</td>
<td>14%</td>
<td>2%</td>
</tr>
<tr>
<td>1984</td>
<td>91%</td>
<td>60%</td>
<td>42%</td>
<td>16%</td>
<td>0%</td>
</tr>
<tr>
<td>1985</td>
<td>80%</td>
<td>34%</td>
<td>47%</td>
<td>15%</td>
<td>1%</td>
</tr>
<tr>
<td>1986</td>
<td>86%</td>
<td>61%</td>
<td>61%</td>
<td>20%</td>
<td>2%</td>
</tr>
</tbody>
</table>
following the order of the choices. Only 1% of the responses thought that there was no need to take care of computers. Furthermore, there were 116 students who selected only one option and out of those, 56% thought that the computers were 'Fast in speed', 7% 'Accurate in calculation', 33% 'Can solve problem for you' and 4% 'Easy to use'. That was no student who selected just the 'No need to take care of'.

4.4.2 Cross-Tabulation Results

Some of the results in the closed-answer questions of this survey will be cross tabulated in this section for analysis. The reader is reminded that part of the purpose of the survey was to investigate any changes in computer awareness of students over these years. Therefore the emphasis will be on a yearly comparison of the results instead of on a class or departmental basis.

The students' computer awareness may be affected directly or indirectly by many factors, such as environment, social background, age, etc. Since the evolution of electronic technology, such as the innovation of the integrated circuit (I.C.), the very large scale integrated circuit (VLSI) and the microprocessor, the cost of hardware products in the computer market has dropped drastically in recent years, especially in terms of value per computer memory byte. The
price of a personal computer is now within the reach of many people, and even students. This provides more opportunities for them to have close encounters with modern technology. The results of this questionnaire confirm this conjecture.

Table 4.14 shows the mean values of the students who thought they knew about computers in the four years according to their age groups. The mean values were calculated by using the six-point scale ranging from zero to five corresponding to from 'nothing' to 'very well' respectively as defined previously. Please note that due to no respondents in the age 24 or above category in 1984, no mean is given. Reading from the table vertically, the mean values of each age group had a positive increment compared with the previous years, with only a few exceptions, such as in 1984. The inconsistency may be due to the effect of incoherent or small size of the sample group. When the table is inspected horizontally, it was found that the mean values also increased in age group. Therefore it may, apart from some abnormalities, be concluded from the calculated mean values that what students thought that they knew about computers for one year is higher than that for the students in the previous years, and a similar phenomenon also appeared if the older age group is compared with the younger age groups. Table 4.15 shows the mean values as regards sex. Speaking overall, male students had a higher score than female students. The result can be explained as that in the traditional social background, it is in general expected that male students are more interested in science,
### Table 4.14  Mean Of Question 3 By Age By Year

<table>
<thead>
<tr>
<th>Year</th>
<th>Below 20</th>
<th>20-21</th>
<th>22-23</th>
<th>24 or above</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>.7857</td>
<td>1.0000</td>
<td>1.6500</td>
<td>1.6000</td>
</tr>
<tr>
<td>1984</td>
<td>1.1053</td>
<td>1.3333</td>
<td>3.0000</td>
<td>-</td>
</tr>
<tr>
<td>1985</td>
<td>.8333</td>
<td>1.4800</td>
<td>1.9667</td>
<td>1.9643</td>
</tr>
<tr>
<td>1986</td>
<td>1.2391</td>
<td>1.3793</td>
<td>2.0909</td>
<td>2.6875</td>
</tr>
<tr>
<td>Overall</td>
<td>1.0606</td>
<td>1.2778</td>
<td>1.9385</td>
<td>2.0313</td>
</tr>
</tbody>
</table>

### Table 4.15  Mean Of Question 3 By Sex By Year

<table>
<thead>
<tr>
<th>Year</th>
<th>Male</th>
<th>Female</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>1.4156</td>
<td>0.5200</td>
</tr>
<tr>
<td>1984</td>
<td>1.5667</td>
<td>0.8462</td>
</tr>
<tr>
<td>1985</td>
<td>1.7294</td>
<td>2.2500</td>
</tr>
<tr>
<td>1986</td>
<td>1.8769</td>
<td>1.1081</td>
</tr>
<tr>
<td>Overall</td>
<td>1.6537</td>
<td>0.9367</td>
</tr>
</tbody>
</table>

### Table 4.16  Cross Tabulation On Question 4 And Question 5

<table>
<thead>
<tr>
<th>Year</th>
<th>Well</th>
<th>Fair</th>
<th>Few</th>
<th>Little</th>
<th>Nothing</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>1(1%)</td>
<td>9(10%)</td>
<td>14(24%)</td>
<td>39(62%)</td>
<td>8</td>
</tr>
<tr>
<td>1984</td>
<td>1(2%)</td>
<td>6(16%)</td>
<td>6(30%)</td>
<td>17(70%)</td>
<td>3</td>
</tr>
<tr>
<td>1985</td>
<td>2(2%)</td>
<td>20(25%)</td>
<td>17(44%)</td>
<td>28(75%)</td>
<td>2</td>
</tr>
<tr>
<td>1986</td>
<td>8(8%)</td>
<td>17(25%)</td>
<td>11(35%)</td>
<td>35(70%)</td>
<td>5</td>
</tr>
<tr>
<td>Total</td>
<td>12(4%)</td>
<td>52(19%)</td>
<td>48(33%)</td>
<td>119(69%)</td>
<td>15</td>
</tr>
</tbody>
</table>
technology and engineering subjects. In the next section, two statistical testing techniques will be applied to test the significance of some of the hypotheses in order to give a more rigorous conclusion.

Table 4.16 gives the distribution on 'How much do you think you know about computer' and 'Have used a computer or an item that is controlled by computer'. The bracketed values are cumulative percentages. Reading from the table, it is obvious that the percentages of students both who thought that they knew about computers and had hands-on experience increased somewhat over the years. The results shown may be due to its being easier for students to obtain and to gain access to computer facilities in these years. The factor of price reduction of microcomputers, mentioned before, also cannot be neglected. However, the increase started to become negative in 1986, starting in the point 2 ('a few') and downward scales. This phenomenon can easily be verified by the analysis shown in the next section.

4.4.3 Analysis on Self-Judgement Questions

There were two questions concerning self-judgement in the survey. The first one, Question 3, was "How much do you think you know about computers?". In this question, a six-point scale ranging from 0 to 5 corresponding to from 'Nothing' to 'Very well' was given. The students were asked
to select any one of the options, without making any comparison with others, that they thought was most applicable to themselves. Table 4.17 gives the number of responses in each scale by year. Since no students thought they knew about computers 'Very well', this option was omitted from the table. Reading from the table, there was no observed score pattern within the data although it is pointed out that the mean value increased somewhat each year. To investigate this, the Chi-square analysis was used to test if their differences were significant. However, for Chi-square analysis, entries with small values in the contingency table can lead to a large error. Investigating the data in Table 4.17, it was found that some of them were too small, particularly in the last column. To lessen the effect due to small numbers, the 'Fair' and 'Well' columns were combined to give one single column of a 'Quite well' score. The contingency table then consisted of 4 columns, being 'Nothing', 'A little', 'A few', and 'Quite well', as shown in Table 4.18. Then each year was being compared with the other years. The null hypothesis being tested was that the proportions of the four categories in one year were the same as for following years. The alternate hypothesis was that the proportions were not the same. Table 4.19 gives the results of the Chi-square analysis.

For a 4 x 2 contingency table, the number of degrees of freedom is three. At a 95% confidence interval, the critical value of the Chi-square distribution is 7.8. Those
Table 4.17  
Response On Question 3 By Year

<table>
<thead>
<tr>
<th>Year</th>
<th>Nothing</th>
<th>Little</th>
<th>Few</th>
<th>Fair</th>
<th>Well</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>22</td>
<td>49</td>
<td>21</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>1984</td>
<td>6</td>
<td>24</td>
<td>6</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>1985</td>
<td>4</td>
<td>40</td>
<td>21</td>
<td>22</td>
<td>2</td>
</tr>
<tr>
<td>1986</td>
<td>14</td>
<td>47</td>
<td>15</td>
<td>18</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 4.18  
Re-grouped Response On Question 3 By Year

<table>
<thead>
<tr>
<th>Year</th>
<th>Nothing</th>
<th>Little</th>
<th>Few</th>
<th>Quite Well</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>22</td>
<td>49</td>
<td>21</td>
<td>10</td>
</tr>
<tr>
<td>1984</td>
<td>6</td>
<td>24</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>1985</td>
<td>4</td>
<td>40</td>
<td>21</td>
<td>24</td>
</tr>
<tr>
<td>1986</td>
<td>14</td>
<td>47</td>
<td>15</td>
<td>26</td>
</tr>
</tbody>
</table>

Table 4.19  
Results Of Chi-square Analysis On Question 3

<table>
<thead>
<tr>
<th>Compared Year</th>
<th>Chi-square Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983 - 1984</td>
<td>2.8</td>
</tr>
<tr>
<td>1983 - 1985</td>
<td>17.6*</td>
</tr>
<tr>
<td>1983 - 1986</td>
<td>9.9*</td>
</tr>
<tr>
<td>1984 - 1985</td>
<td>7.8*</td>
</tr>
<tr>
<td>1984 - 1986</td>
<td>1.9</td>
</tr>
<tr>
<td>1985 - 1986</td>
<td>5.5</td>
</tr>
</tbody>
</table>

* Statistically significant
Chi-square values marked with an asterisk are significant. That is, comparing between 1983-1985, 1983-1986, 1984-1985, the null hypothesis is rejected at the 0.05 level of significance, which shows there is a statistical likelihood of the students rating their knowing about computers differently in these years.

To see why the differences for the other three years were not significant, the sample classes were examined. In the year 1984, due to availability of survey classes and limited resources, there was only one class of students in the Department of Mathematical Studies included in the sample survey and only a total of 43 respondents were obtained (see Tables 4.1 and 4.2). If this was to be compared with the large sample size of other years, some kind of bias should be expected. This might be the explanation for a low Chi-square value in 1983-1984. From the same token, it might also be part of the reason why the Chi-square value for the years 1984-1986 was not significant. The marginal Chi-square value in 1984-1985 compared with that of the distribution critical values reinforced this proposition. However, if another contingency table was to be constructed to make comparison with only the students of Department of Mathematical Studies in 1983-1984, a Chi-square value of 35.03, which was highly significant if compared with the distribution value of 7.8, was obtained. This result might suggest that if a large sample size had been collected in 1984, homogeneous Chi-square values over all these years may
have occurred. Another approach to analyze how the significance of hypothesis from the mean value difference is applied is as given in Lindgren [10]. The Z-test analysis was used to test if there was significant difference between population or large sample size means. The mean values and standard deviations of the students who thought they knew about computers given by years as defined previously was calculated and is shown in Table 4.20. Then each year's mean was being compared with the means of the other years. The null hypothesis being tested was that there is no difference in mean increase between what students thought they knew about computers in one year and that in the other years. The alternate hypothesis was that there was a difference in mean increase. Table 4.21 gives the results of this analysis.

Since only two mean values were compared, the number of degree of freedom is one. At a 95th percentile confidence, the critical Z-value is 1.65. Those Z-values marked with an asterisk are significant. That is, comparing between the years 1983-1984, 1983-1985, 1983-1986, the null hypothesis is rejected at the 0.05 level of significance. The result shown was almost the same as that of the Chi-square test except in 1984-1985, where the Z-value differed by only about 0.11 from the critical value, but was covered in the 94th percentile confidence. The negative Z-value in the compared years 1985-1986 showed a decrease in mean values between these two years.
Table 4.20  
Results On Knowing About Computer

<table>
<thead>
<tr>
<th>Year</th>
<th>Mean Values</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>1.1961</td>
<td>0.9124</td>
</tr>
<tr>
<td>1984</td>
<td>1.3488</td>
<td>0.9731</td>
</tr>
<tr>
<td>1985</td>
<td>1.6147</td>
<td>0.9119</td>
</tr>
<tr>
<td>1986</td>
<td>1.5980</td>
<td>1.1626</td>
</tr>
</tbody>
</table>

Table 4.21  
Results Of Z-test Analysis

<table>
<thead>
<tr>
<th>Compared Year</th>
<th>Z-Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983 - 1984</td>
<td>1.85*</td>
</tr>
<tr>
<td>1983 - 1985</td>
<td>3.33*</td>
</tr>
<tr>
<td>1983 - 1986</td>
<td>2.75*</td>
</tr>
<tr>
<td>1984 - 1985</td>
<td>1.54</td>
</tr>
<tr>
<td>1984 - 1986</td>
<td>1.33</td>
</tr>
<tr>
<td>1985 - 1986</td>
<td>-.12</td>
</tr>
</tbody>
</table>

* Statistically significant
Results shown in the two significance tests suggested the same idea that it was statistically significant that students increasingly thought they knew more about computers in the years 1983 to 1985. This synchronized with the booming of the microcomputer market in the early 1980s and the introduction of the Computer Studies Pilot Scheme in the secondary schools in 1982 in Hong Kong. Although the first batch of students also completed the Computer Studies Scheme did not appear before 1984, it has promoted a common interest in computers among the students ever since. As more students were aroused by the new developments of electronic technology in the microcomputer world, so it became obvious in the statistics of the students in the Hong Kong Polytechnic. The fact that figures in 1986 did not show a significant effect might be due to the levelling out of the general computer awareness in students in these years. As it will be understood, it is not possible to assume that the amount they thought they knew about computers would increase indefinitely. The author believes that the increase may reach a point of equilibrium and that there will not be another serious significant change in respect of this in forthcoming years.

In Question 4, another self-judgement question "Have you ever used a computer or an item that is controlled by computer?" was asked. The response given by year is shown in Table 4.22. the Chi-square analysis was then performed in a manner similar to that outlined previously. The null
Table 4.22  Response On Question 4

<table>
<thead>
<tr>
<th>Year</th>
<th>Yes</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>71</td>
<td>31</td>
</tr>
<tr>
<td>1984</td>
<td>33</td>
<td>10</td>
</tr>
<tr>
<td>1985</td>
<td>69</td>
<td>20</td>
</tr>
<tr>
<td>1986</td>
<td>76</td>
<td>24</td>
</tr>
<tr>
<td>Total</td>
<td>249</td>
<td>85</td>
</tr>
</tbody>
</table>

Table 4.23  Results Of Chi-square Analysis

<table>
<thead>
<tr>
<th>Compared Years</th>
<th>Chi-square Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983 - 1984</td>
<td>0.66</td>
</tr>
<tr>
<td>1983 - 1985</td>
<td>1.72</td>
</tr>
<tr>
<td>1983 - 1986</td>
<td>0.90</td>
</tr>
<tr>
<td>1984 - 1985</td>
<td>0</td>
</tr>
<tr>
<td>1984 - 1986</td>
<td>0</td>
</tr>
<tr>
<td>1985 - 1986</td>
<td>0.12</td>
</tr>
</tbody>
</table>
hypothesis being tested was that the hands-on computer experience in one year was the same for following years. The results obtained are given in Table 4.23.

As defined, the contingency table is $2 \times 2$ and the number of degree of freedom is 1. At the 0.05 level of significance, the critical Chi-square distribution is 3.84. There is no Chi-square value which is significant. Therefore, it was suggested that although it appeared that there were some changes in the data, there was no significant difference in hands-on computer experience of students in these years.

4.4.4 Computer Ideas Questions

There were three questions concerning the computer ideas that the students had. Question 5 was a multiple-choice question and Questions 8 and 9 were open-answer questions. These questions were so selected to see if the students had a general ideas of the present computer world.

Question 5 asked "What advantage(s) do you think a computer has?". Five choices were given and not all choices were correct. In option a, it was given "Fast in speed". There is no doubt that computers can work at a tremendous speed. Starting from the ENIAC which could perform 5,000 arithmetic operations a second, the speed of computers has since
increased enormously. The term MIPS (Million Instructions Per Second) was introduced only a few years ago. But right now, there are already people talking about BIPS (Billion Instructions Per Second) and TIPS (Trillion Instructions Per Second). There is research work taking place on the Photonic Computer, which is in a different family from the Electronic Computer, and which may be able to operate at the speed of light. It seems that the increase in speed will never come to an end. In Option b, "Accurate in calculation" was given. This is, however, not an advantage a computer has. Numerical values stored inside a computer are normalized and converted either in binary form, hexadecimal or other forms, depending on the design of the system. However, no matter what form is being used, all stored numbers have to have a finite number of digits. Some may have 8 digits, some 10, some may have 16 or more. As long as only a finite number of digits is possible, errors are expected to occur in representing some numerical values. For instance, all the irrational numbers, such as square root 2; some fractions, such as one-third; cannot be expressed in a finite number of digits. Some minicomputers give only 6 decimal places in single precision calculations, far less in accuracy than even a hand-held calculator. If the computer were accurate in calculation, most elementary Numerical Analysis text books would not have a chapter on "Error Propagation" in the first place. Research on symbolic computation has been done in the hope that calculation errors introduced by computers can be reduced. In Option c,
it was given "Can solve problems for you". This is also not true either. A computer can only do things under pre-programmed instructions. It is being used to provide answers to problems not because it can solve them, but because the algorithms and the logic behind the problems are known and have been fed into the computer. Then the problem is handled accordingly if correct input format and data is supplied. Without the given pre-programmed solution, the computer is completely ignorant. In Option d, it was said that the computer was "Easy to use". In the early stage of the computer era, if a job was to be submitted and run, a complete team of experts to supervise the operations of the computer was necessary. Nowadays, when a user sits in front of a computer terminal, there is also a group of systems analysts, programmers and operators supporting his job in the computer centre. Someone might think that this does not apply to microcomputer users. However, if one wants to use the microcomputer effectively and efficiently, not to mention that one needs to know the required computer language, one also needs to study the computer's system manuals, reference manuals, etc. If counted together with the other optional manual texts, these may stack as high as, if not exceed, the computer itself. Some well written software packages run on microcomputers are claimed to be user friendly and manual driven, but inevitably, the more powerful the software package is, the more complicated it is. To understand the options and switches provided by a sophisticated software package would take weeks of hard
study and trial and error processes. Furthermore, most communications with the computer have to go through the randomly organized keyboard which cripples many users. It is hoped that the new generation of computers will be much easier to use by at least accepting instructions from human dialects. The last option, Option e, was that there was "No need to take care of" computers. In fact, it is the contrary. One common weakness of all electronic equipment, compared with other types such as mechanical ones, is that they can only function in a limited environment, such as a specified narrow temperature and humidity range. They will not work if it is too hot, too cold, or too wet. Direct contact with liquid will short the circuits. The climate in most regions of the world, such as Hong Kong, would not let the computer survive without a controlled environment. Some computers, like the super-conductor computers, even require a built-in cooling system to dispense the heat generated while operating. Their sensitivity in electrical power also needs extensive care. Although usually they do not consume much power, a small fluctuation of voltage will cause unpredictable results -- data may be lost, or the equipment may even be damaged.

From these five options, only option a is acceptable, but the students could tick any number of choices ranging from only one to all five. Taking all the combinations from these 5 options, there were in total thirty-one cases. But not all the cases were included in the responses. There
were ten empty entries to this question, out of which combinations including the option e had a nine counts share. Table 4.24 gives the first ten most commonly chosen combinations of options ranked by overall responses compared with the ranks of the four years. The number of responses is given in brackets next to the rank.

Despite of what advantages computers do actually have, as discussed above, the students did show a similarity in their concepts of computers over these years. That is, the students not only thought that computers were fast, but also could give accurate calculations and could solve problems. This brings to the attention of the author in the development of the Computer Assisted Learning software package the need to put some emphasis on the advantages and disadvantages of a computer.

In Question 8, the students were required to "Name 5 computer languages" and in Question 9 to "Name 5 computer manufacturers, give 1 computer model for each". Since these two were open-answered questions, analysis applying numerical techniques would be somewhat difficult. However, after taking tally, some results are given as follows. Tables 4.25 and 4.26 give respectively percentages of students that could name three or more computer languages and the first five common computer languages given by students. As seen from the tables, there was a noticeable increase in percentages between 1983 and subsequent years
Table 4.24  
Comparison In Rank In Question 5

Combinations in Options          Overall  1983  1984  1985  1986

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>abc</td>
<td>1(71)</td>
<td>1(24)</td>
<td>3( 8)</td>
<td>3(11)</td>
<td>1(28)</td>
</tr>
<tr>
<td>a</td>
<td>2(65)</td>
<td>3(16)</td>
<td>2( 9)</td>
<td>1(26)</td>
<td>3(14)</td>
</tr>
<tr>
<td>ab</td>
<td>3(62)</td>
<td>2(19)</td>
<td>1(12)</td>
<td>2(12)</td>
<td>2(19)</td>
</tr>
<tr>
<td>c</td>
<td>4(38)</td>
<td>4(12)</td>
<td>4( 4)</td>
<td>2(12)</td>
<td>4(10)</td>
</tr>
<tr>
<td>ac</td>
<td>5(33)</td>
<td>5( 9)</td>
<td>5( 3)</td>
<td>2(12)</td>
<td>5( 9)</td>
</tr>
<tr>
<td>abcd</td>
<td>6(21)</td>
<td>6( 7)</td>
<td>5( 3)</td>
<td>5( 2)</td>
<td>5( 9)</td>
</tr>
<tr>
<td>abd</td>
<td>7(11)</td>
<td>7( 3)</td>
<td>5( 3)</td>
<td>5( 2)</td>
<td>6( 3)</td>
</tr>
<tr>
<td>b</td>
<td>8( 8)</td>
<td>6( 7)</td>
<td>( 0)</td>
<td>6( 1)</td>
<td>( 0)</td>
</tr>
<tr>
<td>acd</td>
<td>9( 7)</td>
<td>9( 1)</td>
<td>( 0)</td>
<td>4( 4)</td>
<td>6( 3)</td>
</tr>
<tr>
<td>d</td>
<td>10( 5)</td>
<td>( 0)</td>
<td>( 0)</td>
<td>4( 4)</td>
<td>8( 1)</td>
</tr>
<tr>
<td>ad</td>
<td>10( 5)</td>
<td>9( 1)</td>
<td>6( 1)</td>
<td>6( 1)</td>
<td>7( 2)</td>
</tr>
</tbody>
</table>

where:  
a - Fast in speed  
b - Accurate in calculation  
c - Can solve problems for you  
d - Easy to use  
e - No need to take care of

Table 4.25  Percentage Of Response On Computer Language Names

<table>
<thead>
<tr>
<th>Year</th>
<th>3 or more</th>
<th>4 or more</th>
<th>upto 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>65%</td>
<td>43%</td>
<td>17%</td>
</tr>
<tr>
<td>1984</td>
<td>71%</td>
<td>55%</td>
<td>34%</td>
</tr>
<tr>
<td>1985</td>
<td>73%</td>
<td>75%</td>
<td>41%</td>
</tr>
<tr>
<td>1986</td>
<td>82%</td>
<td>66%</td>
<td>26%</td>
</tr>
</tbody>
</table>
Table 4.26  Percentage Of Response On The Five Languages

<table>
<thead>
<tr>
<th>Year</th>
<th>Assembler</th>
<th>BASIC</th>
<th>COBOL</th>
<th>FORTRAN</th>
<th>Pascal</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>14%</td>
<td>82%</td>
<td>73%</td>
<td>70%</td>
<td>35%</td>
</tr>
<tr>
<td>1984</td>
<td>26%</td>
<td>91%</td>
<td>81%</td>
<td>67%</td>
<td>44%</td>
</tr>
<tr>
<td>1985</td>
<td>31%</td>
<td>93%</td>
<td>81%</td>
<td>90%</td>
<td>79%</td>
</tr>
<tr>
<td>1986</td>
<td>9%</td>
<td>90%</td>
<td>76%</td>
<td>87%</td>
<td>62%</td>
</tr>
</tbody>
</table>

Table 4.27  Percentage Of Response On Computer Companies

<table>
<thead>
<tr>
<th>Year</th>
<th>3 or more</th>
<th>4 or more</th>
<th>upto 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>93%</td>
<td>83%</td>
<td>48%</td>
</tr>
<tr>
<td>1984</td>
<td>63%</td>
<td>40%</td>
<td>35%</td>
</tr>
<tr>
<td>1985</td>
<td>74%</td>
<td>55%</td>
<td>45%</td>
</tr>
<tr>
<td>1986</td>
<td>63%</td>
<td>46%</td>
<td>31%</td>
</tr>
</tbody>
</table>

Table 4.28  Percentage Of Response On Computers

<table>
<thead>
<tr>
<th>Year</th>
<th>APPLE</th>
<th>ATARI</th>
<th>IBM</th>
<th>PDP or VAX</th>
</tr>
</thead>
<tbody>
<tr>
<td>1983</td>
<td>65%</td>
<td>27%</td>
<td>53%</td>
<td>48%</td>
</tr>
<tr>
<td>1984</td>
<td>72%</td>
<td>33%</td>
<td>49%</td>
<td>9%</td>
</tr>
<tr>
<td>1985</td>
<td>82%</td>
<td>19%</td>
<td>88%</td>
<td>25%</td>
</tr>
<tr>
<td>1986</td>
<td>70%</td>
<td>18%</td>
<td>80%</td>
<td>33%</td>
</tr>
</tbody>
</table>
that students could name three or four or five computer languages, and it is also evident that BASIC and other high level languages were gaining more renown. Table 4.27 and Table 4.28 show respectively the percentages of students who could name three or more computer companies and the percentages of four most common computers. Response on being able to name the computer manufacturers seemed to be declining in the four years, but the popularity of individual computers showed some interesting trends. In 1982, when the Computer Studies Scheme was first introduced, the first batch of microcomputers supplied by the Hong Kong Education Department was Atari, therefore this microcomputer was brought more to the attention of the students. In 1984, when APPLE computers were on the school list, Atari gave way and its popularity was replaced by the APPLE computer. As the IBM microcomputer is getting its share in the personal computer market, it gained the students' recognition in that it led Atari by as much as 69% in 1985. The fact that DEC's computer was known to students might be due to the Hong Kong Polytechnic using VAX and PDP minicomputers so that students had a greater chance of knowing the name from sources like the prospectus or friends in the Polytechnic. The results obtained were expected and self-explanatory.

There are three other observations obtained from these two open-answer questions. The first was that occasionally there were incorrect spellings in computer language names. Actually, most language names are acronyms of words, such as
FORTRAN, coming from FORMula TRANslation, and COBOL from COmmon Business Oriented Language. If the students had known where the names come from, they would not have got them wrong. Secondly, some software packages, such as dBase III and WORDSTAR, were mistakenly given as computer language names. This clearly showed lack of understanding of the differences between the two. Lastly, some students had mixed-up the computer company names with the computer model names, for instance, some students did not realize that DEC is the company's name and the computers they developed are modelled PDP and VAX.

Nevertheless, the questions asked concerning computer ideas were not meant to be a thorough test and were by no means representative. They were so chosen because it was found that these concepts sometimes seemed to be misunderstood by some students and it is the author's interest to identify some of them and to bring them to the attention of the students. A brief discussion of these questions with the students was given every time after the survey, their interest in wanting to gain more computer knowledge was conspicuous from the questions raised and the excitement shown on their faces.

4.4.5 Their Computer World

With all these modern electronic technology inventions,
students could well have their own ideas and expectations about this restless computer world. To find out what kind of computer world they did have in their minds, two open questions were asked. Question 6 stated that "An ordinary person has I.Q. around 100. What I.Q. score you would give to a computer?". Table 4.29 summarized their response. Over 44% of the response gave a zero score in Intelligence Quotient to computer and just above one-tenth of them
thought that a computer has an intelligence similar to that of an ordinary person. There were thirteen students who did not give a score to the computer but commented either "Cannot be compared with" or "Depends on how sophisticated is the operating system and on the intelligence of the user". One of the largest topics in research and development other than making computers faster, smaller in size and larger in memory capacity, is the artificial intelligence, which was restricted to large computers a few years ago and can now be available to people who are using microcomputers [9]. In the future generation of computers, it will be possible to have functions such as pattern recognition, knowledge representation, decision making, learning, etc. as basic criteria which can carry out tasks that would now require human intelligence.

In question 7, it was asked "What things you would like to be processed by the computer?". Although there were extreme answers like "Everything", or "Nothing", or simply "All assignments given by lecturers", there were some common interest things that they wanted to be done by the computer. Among all their various wishes, the first priority was to have the computer to do what it was originally envisaged for -- the number crunching work. More than half of the students in each year's response wanted their tedious computations, routine calculations and repeated iterations to be done by the computer, some hoped to have their data analyzed as well. Their second wish in
the list was to let the computer handle their records, such as their friends' telephone numbers and addresses. This can be viewed as elementary thinking of data processing and a data base system. It is also worthwhile to point out that only in 1985 and 1986, were there responses of hoping that report typing, word processing and the like could be done using the computer. Students might be aware of the impact of office automation in the commercial society that in turn widened their thinking. Other than serious work, there were approximately 10% to 20% in each of the four years that associate game playing with computers. There is nothing wrong with this, after all, most games are logical, or mathematical, and stick to rules, they provide good analogies to actual situations involving human beings and their environment. Business, military and pilot training often apply game theory with the computer to simulate the real operation which may be costly, dangerous or even impossible.

4.5 Conclusion

Computer literacy is a dynamic area of education and training. The technology and its applicability are constantly breaking new barriers. This knowledge can be regarded as an everyday skill available to everyone. Like all other skills, it decays if it is not taught and learnt thoroughly, and if it is not practised and updated from time
to time. This process involves repeatedly the three procedures of knowing, understanding and doing. Studies have shown that computer literacy enables students to cope confidently with the technology when they encounter it in its various forms at work and at home, and to adopt a considered view of its social implications. It surely enhances their future employment potential in all respects. A computer literate student does not need to be an expert in program writing, but at least he should fulfil the two statements as defined [11] that he should have:

"(i) A general awareness of Computer jargon and computer systems
(ii) The skill to react with a computer to perform useful tasks."

To give a conclusive accomplishment to students of Hong Kong over the preliminary stage of computer literacy seems too early and inappropriate at the moment. However the aims and objectives are there and some elementary positive results are evident. The survey on computer awareness of Hong Kong Polytechnic students in these years has confirmed this fact. In the survey, it has been shown that it is statistically significant that the Hong Kong Polytechnic year one students have increasingly thought that they knew more about computers since 1983. A less significant value compared by the critical Chi-square value in a 95% confidence interval only appeared in the year 1986 when the
author believes that the significance level would not increase indefinitely and that the students had reached an equilibrium point of thinking that they knew about computers. However, in responding to the question whether they had ever used a computer or an item that is controlled by computer, the null hypothesis which assumed the proportions of response in one year were the same for the following years cannot be rejected. This may suggest either there was not much change in students' hands-on computer experience or the students were not aware that many of the machines encountered daily, such as banks' electronic teller cash machines, subway gates and many home appliances run under the operation of computers or microprocessors.

In the question concerning the advantage of computers, most students agreed that computers were fast. But many of them did not realize that computers were not accurate in calculation and could not solve problems. Their misunderstanding may be due to the processes involved in learning mathematics. Generally they were too often presented with precise or integral data to which a particular mathematical technique was expected to be applied, and no question would be raised as to the degree of accuracy of the result, it is only either right or wrong. They were seldom exposed to irrational numbers or transcendental numbers which could not be represented exactly by computers. It is also unlikely that they would have encountered data that had no simple existing algorithm
to handle it. Therefore they had the illusion that everything fed into the computer would produce an answer which must be perfect. Not arguing from the pedagogic's points of view, sometimes it might be worthwhile to give guidance to students to let them exploit and experience the powerfulness of the computer, but at the same time perceive its limitations.

From the survey, most students were able to name three or more computer languages and computer models. Although there were occasional misspellings and mixing up of names, they had shown adequate computer literacy as defined in [11]. The computer world in the students' mind was practical too. Many of them thought that computers had an intelligence score of zero but would like the computer to manage their academic affairs, their records and to play games with them. They would like the computer to be their personal assistant as well as their good companion.

This survey should, by no means, be viewed as a comprehensive study in computer literacy of Hong Kong students. It is only in the interest of the author to find out what computer backgrounds, in particular, do the first year Hong Kong Polytechnic students have and whether there was any significant change in their awareness of computers since the Computer Pilot Scheme was introduced in Secondary Schools in 1982. The findings were found to be interesting and fruitful. Also, based on the results, the author has
been able to decide the topics in the Computer Assisted Learning software packages, which will be discussed in the next few chapters, most suitable for the year one Hong Kong Polytechnic students taking the Numerical Analysis course.
CHAPTER 5

A CAL Package Used By Students Of HKP

5.1 Introduction

CAL has been used in the education sectors for some time now in many countries, such as the United States [12]. In France, a study to develop strategies and curricula on teaching using minicomputers was undertaken in the early 1970’s. Later, when the impact that microcomputer technology would have on the overall computer world was realized, a shift of emphasis to microcomputer systems was quickly made [15]. Nowadays microcomputers are not only used substantially in secondary schools all over the world [18], they are also used heavily in primary schools [19]. Pupils use microcomputers to help their skills in mathematics as well as spelling. The Turtle and BBC Buggy [20], robots designed for children’s use, are examples of the used microelectronic technology to develop and stimulate the problem-solving skills of primary school students.

The thing that most concerns educationalists is the psychological effect that would be caused by using CAL. The supporters of behaviorism see CAL as the implementation of
programmed learning. The computer, which operates at high speed and allows versatile interaction by branching, takes over the mechanical teaching machine to make the individual "frame" appear on a colourful television screen. Skinner, a psychologist and a behaviorist, was primarily interested in conditioning which is based on Thorndike's law of effect of stimulus versus response, and had his own version of "operant conditioning" [23]. He believes, through experiments with rats and pigeons, that learning is a behaviour process of exploration and trial and error and that the behaviour pattern would change dramatically if reward is obtained. Furthering his experiments, Skinner was able to discover four basic factors [26, P. 21] that CAL should be employed to increase learning effectiveness. These four factors are summarized as follows:

(a) Piecewise but connected sequential information
(b) Constant active response
(c) Immediate feedback and reward
(d) Progress at one's own pace

This "reinforcement theory" has been used by many educationalists who claimed successful outcomes were obtained. However, there are some criticisms on unwise use or badly designed CAL packages. Amongst the most influential and interesting at the present time is Papert [39] in Mindstorms. He criticizes much of what is done at present with computers in schools in that the computer is
now programming the children how and what to learn instead of being programmed itself by the children. He further provides an amended, broad and theoretical way of treatment of the use of computers for learning. He introduces the idea that children are able to learn programming by controlling the movement of the now renowned Turtle -- a floor robot which can be made to leave a trace of its travels. At the heart of his idea is that once the children begin to explore the world of lines, shapes, space and the relationship between them, they will develop the adult-like formal logical thinking, abstract reasoning, and imaginative hypothesis. As Papert himself puts it,

"children can learn to use computers in a masterful way, and ... learning to use computers can change the way they learn everything else."

As discussed so far, the use of computers in an educational setting falls into two categories: teaching with the computer and teaching about the computer. In Hong Kong, teaching about the computer in the secondary school was started more than four years ago. The Computer Section of the Education Department is working on the implementation of LOGO, a sophisticated programming language as well as a theory of thinking, in the lower form as a computer literacy curriculum. However the use of microcomputers in teaching, such as CAL, and particularly in mathematical education, is
very new. The traditional chalk and board still reign over most of the students' learning process. The author, having been teaching in the Department of Mathematical Studies in the Hong Kong Polytechnic (HKP) for a number of years, feels that the idea of using CAL with microcomputers on mathematics subjects should be introduced. He, therefore, having selected a topic, developed the software packages which were then used by the students of the HKP, and is now and in subsequent chapters going to discuss the experience and the findings obtained.

5.2 The Aim and Content

The aim of this work was to develop a CAL software package in mathematics subjects, to introduce it to the students of HKP and to obtain feedbacks after they have been using the package. The feedbacks are then analyzed, results being obtained and conclusions reached on the use of microcomputers in mathematics teaching.

In the first instance, the content material has to be decided. The author has been teaching calculus to engineering students for a number of years, experience having been gained on both what are the weakest areas of the students and what is the best possible teaching approach on some of the topics. In fact, taking into account the advantage of the graphic capability of microcomputers, some
parts of the calculus syllabus, such as polar coordinates and conic sections that many students find difficult, can well be presented by CAL. However, in recent years the author has taught mainly Numerical Analysis, and it seems natural to think that number crunching is the job of a computer and that numerical subjects can be presented magnificently by microcomputers as well as graphic functions. Furthermore, there are quite a few papers published internationally in this subject area by many mathematicians which would provide valuable help and guide lines [44]. In addition, the results shown in the survey of computer awareness of HKP students (see Chapter 4) reveal that over one half of the students thought that the computer had the advantage of performing computations very accurately. Therefore the author decided to develop a CAL package on what and how error would occur when using the computer in Numerical Analysis.

Having selected the topic to be used initially, a choice had to be made as to what type of educational program was to be written. Although Burke [45] gives much finer categories of teaching programs according to their functional, physical and logical designs, it is generally accepted that there are three type of CAL lessons, the drill and practice, the direct instruction/tutorial, and the electronic teaching aid. The nature of CAL software being produced will now be considered using these three types as a basis for discussion.
Drill and practice programs form one of the oldest designs of CAL lessons and have been subject to considerable criticism. The basic role of such a program is to use the computer to display a series of problems of a given type and have the student to solve them one by one. The problems may be in increasing difficulty level and allow multiple attempt answers. Some educators feel that drill and practice, while having been shown to be very effective and to have significant success in improving grade level, is not desirable in our current education system because it provides no more than a mechanism for factual recall and a test of a particular skill.

The second type is the direct instruction/tutorial program. This often involves the delivery of knowledge or concepts by breaking a lesson task down into a series of sub-tasks and the receipt of feedback by interaction of the keyboard from the learner at each stage with the help of graphics. This type of CAL is seen to be a versatile learning aid for individuals with endless applications.

The third category is to use the program solely as an electronic teaching aid. It serves more or less as an electronic blackboard to demonstrate a particular phenomenon or to perform a special experiment which may not be easily explained or done by, say, an overhead projector, a blackboard or in the real life situation. One of the advantages of using the computer is that the process can be
done repeatedly with or without varying the parameters. The flying simulator has been used in this respect successfully for many years.

Among these three types of CAL software styles, it was decided the second one, that is the direct instruction/tutorial program, was to be adopted due to its suitability for the topic "Error". The objectives and the contents of the lesson were then planned.

The primary objectives of the software are:

- To identify what types of errors, in general, may occur when using the computer for numerical applications and why they happen.

- To demonstrate, by using examples and interacting with users, how the errors start and how to minimize them.

- To add interest and to motivate the user in this subject area.

The contents of the topic were planned to include five parts. The first part was the introduction, whilst the second, third and fourth described the three general sources of errors, and the fifth part was to be the estimation of propagation of errors.
In the introduction, the way in which numbers were stored inside a computer was explained. It shows that error was bound to occur when the number of digits used by a computer was finite whereas many numerical values, such as the transcendental number PI, the irrational numbers such as the square root of 2, and some fractions like 1/3, were not.

The second part of the lesson explained that error could be due to the nature of the data and data transfer. The obtained data might come from experiments or observations where accuracy of the instruments and uncertainty of measurements had not been taken into account. Furthermore, when data were keyed in, copied or transferred either manually or electronically, the possibility of inducing errors during the process increases. Some suggestions to minimize such errors were mentioned.

Rounding and truncation errors were demonstrated next in the third part of the lesson. The First Principle of Calculus was given as an example. According to the theorem, in order to obtain a better approximation of the derivative, the values of the interval used have to be very, very small (approaching zero, in calculus jargon), however, in practice, it is only true up to a certain degree. The result would get worse if the interval used is too small. The reason is due to rounding error. This type of error was further elaborated and shown by examples.
The third source of error, errors due to inappropriate interpretation of formulae, was introduced in the fourth part. Two examples were used to demonstrate two different arrangements of the same formula, showing that altering the sequential order of doing the arithmetic would yield different answers. At the end, Horner's Rule, a factorization technique for polynomials based on synthetic division, was explained. This simple grouping of polynomials is one of the methods of rearranging formulae to reduce errors of this type.

The last part of the lesson was to derive the four equations to estimate errors when using the four arithmetic operations. If two values which already had errors were being added, subtracted, multiplied or divided, the propagation error of the sum, the difference, the product, or the quotient can then be approximated. However, due to reasons which are to be discussed later, it was decided not to include this in the final version of the CAL package.

In the initial planning stage, one of the things to be ensured is that adequate examples would be included in each part of the lesson and hands-on trials would be added whenever possible and appropriate. A handout, which covered the lesson material in more detail, would also be prepared to incorporate with the CAL package. A copy of this is included in the Appendix D.
5.3 The Approach

It is obvious that the first consideration of using a CAL package is access to the computer. There were quite a few microcomputers available on the market at the time the work was started but each had its own characteristics and these were not compatible to each other. This made the actual design of the CAL lesson confined by the hardware configuration, such as the graphic capability, the resolution, and, the static, as well as the dynamic, instruction set. Programming in one machine can be completely different from that in another machine. Therefore the choice of microcomputer being used to develop a CAL package is crucial. Once one machine has been decided upon, a change of machine almost means starting the job all over again.

In the author's case, to decide which microcomputer was to be used was straightforward because the APPLE computer was the only microcomputer facility available in the Department of Mathematical Studies at that time. The only thing left to be considered was whether to use a computer language or to use an authoring language to develop the package. If the CAL lessons are to be written in one of the so-called high level computer languages such as FORTRAN, BASIC, PASCAL etc., then the task is reasonably straightforward in the sense that these computer languages have a clear, precisely defined structure and words of command which can be used in
the predefined way. Such languages give complete flexibility and can tailor-make the CAL version to contain whatever feature one wants to include. The only drawback of using computer language is that one has to have a good knowledge of the language and be fully conversant with computer logic. If the CAL lessons are to be written in an authoring language, a further investigation is necessary.

5.3.1 Authoring Language

An authoring language for CAL is a special purpose software package designed to perform specific functions. They are usually easier to learn to use and are quite powerful so that a lot of development time is saved compared with when using a general purpose computer language. This primarily aids the generation process of creating computer assisted learning lessons for teachers who are not programmers and are not fully conversant with computer logic. On the other hand, an authoring language is not as flexible as a programming language and is restricted and has limitations in many ways.

There are a number of authoring languages available on the market, but after some investigation, it was decided to use the authoring language SuperPILOT in the development of the CAL package. SuperPILOT is the second version of the Common PILOT (Programmed Inquiry Learning Or Teaching) language
which was developed in the early 1970's in an attempt to provide teachers, who may know only little about computer programming, with the medium to produce computer based instructional packages. The APPLE SuperPILOT consists of a standard instruction set which is quite mnemonic. There are twenty-six instructions to be used in the four editors: the Lesson Text Editor, Graphics Editor, Character Set Editor, and the Sound Effects Editor. Most of the instructions consist of simply one or two characters which contain three basic components: the instruction name, a colon, and the object of the instruction. For example:

T : Welcome to Lessons on Numerical Analysis

is the "Type" instruction of the SuperPILOT which causes the welcome message to be displayed on the screen. Note that the "T" is the instruction keyword for "Type" which must be followed by a colon and can be either in upper or lower case.

In order to provide some decision making and intelligence over the instruction codes, SuperPILOT offers two ways to alter the execution of most of the instructions. This involves the addition of eight modifiers and five conditioners. The modifiers, if included to be used with the twenty-six instructions, can modify the way in which an instruction operates. For example, "A :" is the instruction code for accepting a response from the user, using the
modifier "S" will give the instruction "AS:" which means to accept only a single character response.

In the case of the five conditioners, they can be used to determine whether or not the instruction will be executed under various elements. For example:

TY : That's right. Well done.

will display (Type) the encouraging remark if the last match is successful (Yes).

Combining these modifiers and conditioners with the instruction set, programs and instructional strategies can be developed into more complex sequences. Furthermore, SuperPILOT is menu driven and prompts are given when working within each editor to provide easy development of a powerful CAL package. Other advantages of using the SuperPILOT are that the special effect instructions provide versatile variation of sound, graphic and colour options which can give an animated display of visual information. Most important of all, in view of Numerical Analysis, it gives a comprehensive mathematics instructions set which includes the algebraic, the transcendental and the trigonometric functions. Other than the complete instruction set, it also provides file management, editors and utilities that help the development of a package at ease. Detail structure, mode of operation and restrictions can be referred to the
manual [46].

One of the disadvantages of using an authoring language is, as it was confirmed later, that the versatility is at the expense of computer time. The software will run noticeably slowly if demanding computation is needed. Unfortunately, in Numerical Analysis, extensive calculation is unavoidable.

5.3.2 The Design and Layout

Many of the principles related to the design and layout of print materials relate to the design and layout of a CAL package. Not all of them apply, however, those that do need to be modified because there are significant differences between the printed page and the computer screen displays. There are ideas and guidelines [49] based on the experiences of practitioners in the field as well as some empirical evidence that certain basic requirements should be sought on the video screen. These suggestions can be grouped into four categories, namely screen format, paging, conciseness and interaction.

The screen format in a package gives a very first impression to the viewer. It is the front door to the presentation of the material. It is suggested that the entire screen should be divided into systematic windows such that the text and graphics are arranged in such a way as to take advantage of
natural eye movements. Text, diagrams, figures and response should be displayed in consistent windows and not jump around all over the screen so that learners do not need to hunt for information they are expected to use. Material on a screen should not be crowded together. If text is given, it would be better to use double line spacing, but whether it is right margin justified or not is unimportant. Hyphenated words and flashing text should be eliminated as this would decrease readability.

The second category is paging. Almost all printed materials are arranged according to pages; this should not be different in CAL lessons. It is recommended that the contents presented should be divided into frames, with each frame being equivalent to a page in a book. If there is any movement or alterations in the graphs, pictures or diagrams, let the participants see the effect of changes. However, do not expect the learner to be able to look at two different things on the screen simultaneously, so any changes should be done one by one and frame by frame. As some people may read faster than others, a fixed computer controlled time for a frame may not be appropriate. Therefore, it is better to let the learner control the duration of the displayed material by making a response on the keyboard, such as "Press any key to continue". This participation of page turning requirement not only can allow users to progress at their own pace but also let them take part in the learning process. It is also helpful to allow readers to page
backward as people sometimes do with written material.

The third category concerning the design and layout of a CAL lesson is the conciseness. Written materials are easy to use and give scope for little ambiguity, but in a computer it may not be quite as simple. It is well known that computers are very fussy in exact matching and preciseness, therefore it is recommended that all instructions given should be clear, simple and concise. It is understandable that notations and symbols are unavoidable, particularly in mathematics subjects. It is suggested that if they are being used, they should be consistent and appropriate to the contents and to the age of the users. However, use of cryptic abbreviations and codes in an attempt to save space is not recommended. Most educationalists do not object to a CAL package making reference to books, tapes, slides etc., as long as these are easily accessible or readily available. In the text, sentence structure should be in accepted spelling and grammar, statements like "GO NEXT PAGE", "TRY SOLVE PROBLEM NOW" should not be used. In checking responses to questions, it is obvious that on the computer it is difficult to check answers to open questions and proofs given by students and these should be kept to a minimum. Answer recognition can be made much easier by multiple choice, tables of solutions or direct numbers. These can also eliminate unnecessary long typing being given to the students and the frustration of giving wrong answers only because of mismatch spellings. Some CAL packages use
graphics and sound as part of their title pages. These would add interest to the user and serve as the motif of the package. But fancy title pages take time to generate and would annoy the users, especially if one uses the program frequently, therefore they should be kept only for a nominal time.

The last thing, but not the least, to observe in producing CAL packages is interaction during the lesson. It is important not to allow the program to present several pages of information without a significant response from the learner. It seems the more interactive a program is the more interesting it is. When responses are being keyed in, the computer should check the validity of the input, for example, numbers should be within range, the data given is correct etc. If the response is incorrect, do not use sarcastic feedbacks such as "Can't you do better than that?", rather, use some encouraging statements like "Nice try, but that is not quite right" and provide some new information or hints to let the learner try again. For a correct response, let the computer give personal and humanized remarks such as "That is right, John. Keep up the good work!".

Much more comprehensive details covering coding hints have been given by Bajpai [50]. Together with the eleven helpful DO's and DON'T's for developing softwares that were gained from experience with the MIME project, they also listed four
basic design features that should be considered in a system as a whole. These features may provide consistency and flexibility of operation combined with confidence that the package so designed can be used with ease.

5.3.3 The Coding

As far as actual coding is concerned, a CAL lesson can be written in one long program or broken up into a number of short programs. Although computer memory is no longer the constraint for a large program that it used to be, several short programs are still more desirable for many reasons. The major advantages are that they are easier to expand or to make changes if needed and they can be tested individually in the development stage hence saving debugging time. The only drawback might be that it would need more frequent access to the disk and would make the overall program length longer due to the overheads taken in passing parameters from one program to the next.

In this CAL 'Error' lesson, the package consists of several program segments, the division being mainly according to the content material. The first and last programs are the 'HELLO' program and 'END' program respectively. The other programs are on the basis of one per topic as laid down in the previous section. Each program started by cleaning the screen, identifying itself and defining the window. This
procedure is recommended in all program development as it saves a lot of trouble in updating and amendment later. There are also some standard subroutines in each program. There is one subroutine to handle requests to go back and review the previous section at any run time and another to be responsible for the linkage of the preceding program. Another common subroutine is to check the validity of the input from the user. A brief outline of each program is being described below.

The 'HELLO' program serves as a starter and gives the greeting and the general introduction to the users. The 'END' program, as its name implies, informs the user of the completion of the lesson. These two programs are somewhat similar and are rather simple at the moment, however, they provide a complete flexibility for further expansion in the future such as the ability to add more modules and to link up with other software.

The second program gives the first and second parts of the lesson contents as explained in Section 5.2. It describes the reasons why error would occur in storing numbers and the first source of induced error. Since the material is fairly straightforward and simple, it is presented as instruction type and included within one program segment.

The third program deals with the idea of the second source of induced error, rounding and truncation errors, mainly by
means of two examples. The two examples both invite the student's participation. The first example, as mentioned before, is the First Principle of Calculus. Four pre-set values, in decreasing interval width, are calculated by the computer to demonstrate the down and up error phenomenon in a given function. Then it invites students to input some interval width and to try for themselves. Results are immediately evaluated and displayed so that students can compare the values of the pre-set intervals with that of their input values. The second example is to sum $\frac{1}{n}$ $n$ times. The program uses the random number generator to choose three numbers ranging from 10 to 50 for evaluation. Due to rounding errors, the sum does not necessarily yield the exact value, that is, one. Students are then invited to try some of their own numbers. It is hoped that, through these two hands-on examples, students will become aware that rounding and truncation errors can happen even in very simple calculations.

In the fourth program, the concept of the third error type is explained. Other than to demonstrate that the order of sequence of doing the arithmetic of the same equation will yield different answers, a special graphic effect is used to illustrate, step by step, how to factorize an n-degree polynomial using Horner's method.

The fifth program simply derives the formulae for estimation of propagation errors. The complete CAL package program
listings, except the fifth program, which was withdrawn later, are included in the Appendix D.

As coding is only one part of the CAL lesson production process, correct coding, design and layout of the package have no guarantee of success. The total CAL package creation process includes several other time consuming processes as well, such as validation, field-testing and constant updating. Various aspects of the use of the CAL package are considered in the next section. Others will be discussed in the next chapter.

5.4 The Use

The CAL package is planned to be used by first year students in the Department of Mathematical Studies in HKP. But before it is actually being used, the software has to be validated by field-testing. The validation, it is believed, should be seen as an integral part of the developmental process and can serve the purpose of improving and refining the CAL lesson. A group of students, who were taking Numerical Analysis and had already covered the topic on Error by normal lecture mode, have been able to react and make suggestions and comments on the package. Their reaction was favourable and they generally felt that the lesson was interesting, easy to understand and not difficult to use. There were comments saying that the package was too slow, too simple and that
some kind of test questions should be included. Their reaction on the material being too simple was expected because they had learned the content already and should not have any trouble understanding it. After all, to make learning simple and easy is one of the aims of CAL.

The relatively slow speed in some parts of the lesson had been noticed during the program developmental stage. As has been mentioned before, the package was written using an author language, SuperPILOT, which took care of all of the management work. The system automatically paging a program by small segments from disk to memory virtually allowed a single program as large as a disk could hold. Together with many other facilities provided by the authoring language, there was no doubt that it would be slow. However, if the program consisted of only standard instructions, general graphics and simple calculations, it ran at normal speed. In fact, the author had to use the 'WAIT' function in some parts of the program to slow down the graphic movements in order to demonstrate the processing effect. Since looping and iterations, which demand extensive computation, are frequently needed in Numerical Analysis, the slowness was evident. Although careful program planning and multiple statements per line had been made to help increase the speed and efficiency of the program, the longest "black out" time observed in the lesson was about five seconds, that was the best the author could do.
The comments that some types of test questions should be included in the lesson were seriously considered and it was decided not to implement these in the package due to the fact that this package was not meant to be a drill and practice one. However, questions to ask the students' understanding on the subject was put in the questionnaire which will be discussed in the next chapter.

Other than some small corrections and improvements, there was one major change over the package due to timing. The running of the whole lesson for an average student was about 45 minutes and the students generally felt that this was too long. Although there was no single strategy of how long a CAL lesson should run, some students may get tired of the computer after five minutes while others work happily for hours and more, it is generally accepted that a half hour or so package is more appropriate. Therefore, after some consideration and discussions with the students, the last part of the lesson was removed in the light that it was somewhat an integral part on its own. Then the total average run time was rightly reduced to a reasonable level.

After the field-testing and the necessary amendments, the package was on time for use for students in the Numerical Analysis class in the Department of Mathematical Studies. They were first year students of the HKP and many of them did not have experience on microcomputers. But the package
was designed to be user friendly and very easy to use. All they needed to do was to borrow the disk from the Mathematics Laboratory, put the disk in the diskette drive and switch on the computer. The lesson would then be loaded and start all by itself. When the lesson was finished, they simply took out the disk, switched off the machine and returned the disk. But before they left, they were required to complete a questionnaire. Nevertheless, just for the sake of some absent minded students, a procedure list together with the incorporated written CAL material was given to them.

Concerning the actual usage of the CAL package, there was a foreseeable technical problem. In the Mathematics Laboratory, there were only two APPLE computers for students' use, but there were 83 students taking the subject. If each student occupied one terminal per hour, it would span one whole week (counting 40 office hours per week) non-stop using the two computers immediately one after another solely for this project. In order to ease the possible congestion, they were given about six weeks' time and were warned well ahead. As it turned out, the operation went smoothly without problem.
CHAPTER 6

Feedback From The CAL Lesson

6.1 The Question Paper and the Evaluation

Even since there has been computer assisted learning, there have been applications in almost every subject of the curriculum. With the advent of the microcomputer, CAL in the classrooms provides for an even wider range of interests [52]. Physical education uses computers to show, for example, body movement in gymnastics. Artists use the computer as a tool to study picture composition, to demonstrate perspective construction, or to create complex patterns. A computer can generate musical notes and, with or without the presence of the teacher, give ideas of scale, chords, keys, time signatures, and so on. Other exemplary application areas are modelling, information storage and retrieval, process control, etc. Of all the subjects of the curriculum using CAL, none has received more attention in this respect than has mathematics. It has gained immense popularity among educationalists and computer scientists in teaching mathematics. The most obvious applications of CAL are in the field of graphics, such as geometry, iterative problems that can arise through the repeated applications of a simple process, and those areas involving complex formulae
and the need to investigate what happens if some of the parameters are changed. Nevertheless, in all the different subjects, it is certain that the use of the computer has led to a review of teaching methods, and in some cases, the subject matter itself.

The design and preparation of a good CAL package is not easy and, if done properly, is often very time consuming. It is estimated that from 200-400 man-hours are required to create lessons totalling one hour in duration [46]. Worse than that, no matter how thorough has been the planning, preparation and production of the software, this will still contain errors, flaws and weaknesses that cannot be predicted. It is unreasonable to assume everything is perfect, especially when it is difficult to know how students will react to any material being used for the first time. Hence the importance of another time consuming task - that of the evaluation of the CAL material.

The main reason for evaluation of a CAL lesson is because even after taking considerable care over the preparation of the material, one can never be sure of the effects and effectiveness of such material. It is only the use of appropriate evaluation techniques to attempt to identify and explain the effect and effectiveness of the software package that can lead to improvements in teaching, and, consequently, in the user's learning. In simple terms, evaluation is to provide information of how well the
material has been communicated to the users.

The process of trying to identify the effects and effectiveness of instruction is generally known as formative evaluation and is concerned with the ongoing process of education. It is not concerned with the grading of students, but with the providence of information to both teachers and students. In this sense, formative evaluation involves an assessment instrument to inform students how they are proceeding during the learning process and not after it. It also makes teachers aware of the difficulties and misunderstandings on the part of students and identifies the areas of weakness within the material that are causing difficulties. This source of information is very important because, resulting from the evaluation, both teachers and students are aware of the areas of difficulty and other weaknesses, and what kind of action is called for. When all the material is investigated, the weaknesses, flaws and errors that occur can be improved and modified. It is suggested that the formative evaluation of the learning material should start very early on, even in the preparation and production stages. It should be a continuous process and is seldom finished if the material is being used year after year.

The evaluation process the author used is identified by two distinct stages. The first stage was a subjective evaluation which was carried out before the CAL packages
were actually used. The second stage was the objective evaluation which was carried out by the students using the package. Under the subjective evaluation, several points were considered. The most elementary step was to eliminate typographical errors. Then the language and wording being used were examined. It is often suggested words and sentences should be plain, straightforward and as short as the subject allows, maintaining, however, accepted spelling and grammar. Other than the language, the tone should be friendly, lively and interesting, and with humour where appropriate. The level of difficulty was also evaluated at this stage. A checklist has been produced by the Open University on the academic acceptability of learning materials. It states that the material should be:

- factually correct
- up-to-date
- adequately supported by evidence
- careful to avoid over simplification or over generalization
- true to the nature of the discipline
- balanced and at pains to present opposing points of view where appropriate.

During the subjective evaluation, a small scale pilot study was carried out. Although this kind of evaluation is a little more objective, which belongs to the second stage, both student comment and performance provided feedback on
all aspects whether it be the material itself, the chosen strategy, the objectives, or the resources and constraints etc. The feedback was seriously considered and a few changes and modifications were made based on the objectives and the theme of a teaching/learning software package through computer aided instruction. There was one major improvement made to the package. The run time of the entire lesson for an average student has been cut to about thirty minutes by reducing the content material. As it was later found out from the objective evaluation, this was a very reasonable and acceptable time for students in running a lesson.

The objective evaluation was to provide information on the effect, effectiveness and successfulness of the learning package. There are general guidelines on what variables should possibly be measured: They include the identification of basic weaknesses in students' prerequisite knowledge and therefore the need to provide remedial tuition, the extent to which each student achieved the objectives using the material, the provision of a record of the students' background and how the learning material is used, and the appropriateness of the instruction style, presentation and built-in interactive questions. The author set out a questionnaire based on these guidelines to evaluate the variables on these aspects of the students using the CAL package. The questionnaire was divided into four sections, namely, the Lesson Content Questions,
Questions On Student’s Particulars, Feedback Questions, and the Open Answer Question Section. In what follows, the responses and results will be discussed in detail. A copy of the questionnaire is included in the Appendix E.

This objective evaluation questionnaire was clearly labelled into four parts, Section A to Section D. Before the students tried to answer the questions, they were required to study the CAL package on Error which is one of the topics in the Numerical Analysis I syllabus. In using the package, the student first had to come to the Mathematics Laboratory and check if there was an APPLE computer available. If there was, then he could borrow the lesson disk from the technician in charge and study the package by himself. The complementary reading material and the questionnaire were given at the same time but the students were requested not to read the questions before the package was completely studied. The package may be used by the students at any time during office hours and a six-week elapse time was given to them to complete the lesson. Despite the fact that there were 83 students sharing the two APPLE computers, no one complained about the limited computing facilities and they managed to finish the study in time.

6.2 Lesson Content Questions

Section A of the questionnaire consisted of five questions
which were all concerned with the lesson content. The questions were:

1. The reason that a stored number may not be the exact value is because . . .
2. Give four methods to minimize error in data transfer.
3. In the example of the first principle of calculus, what h value did you try? How does the error compare with others?
4. In the example of summing \( \frac{1}{n} \) n times, what values of n does the computer use? Which have zero error?
5. Factorize the following using Horner’s Rule:
   \[ x^4 - 3x^2 + 2x - 4 \]

These questions were not meant to be post-test problems although they looked as though they were. They were only treated as a kind of students’ self-assessment questions which were provided as the confirmation of knowledge in the learning process or as the detection of which portions of the learning sequence need additional remedial options. These questions would not be graded in the sense that they would affect the student’s final examination score of the Numerical Analysis I result. The students could find many of the solutions in the complementary reading material and they could refer to it when they felt this to be necessary. Out of the five questions, the first two questions were
straight factual concepts. In the first question, the reason that a stored number may not be the exact value is because numbers stored inside a computer have to have only a finite number of digits whilst most numerical values do not. For the second question, there were five methods mentioned in the lesson to minimize the error in numerical data transfer, they are double check, parity check, range check, consistency check and check digit. Students could choose any four for their answers. In answering these two questions, all the students were able to give the correct solutions. A few of them even elaborated the points in more detail and gave one or two examples. This definitely showed their competence over the content material.

The CAL package demonstrated rounding error with two examples, both of the examples requiring the student’s participation. The first example defined the first principle of calculus which stated that the interval \( h \) should approach zero in order to obtain a better approximation of the derivative of a function. The package, having given a function, preset some values of \( h \), evaluated the derivative of the function, and printed out the errors. The values of \( h \) used by the computer were 0.01, 0.001, 0.0001 and 0.00001. It showed that the error first improved for decreasing values of \( h \) but then got worse as \( h \) became too small. The package then invited the students to try some of their own values.
Question 3 in the questionnaire asked what values they tried and how the errors compared with those when other values were used. In their responses, the students reported using a very large range of numbers, from values near 1 to one ten-thousand-millionth. They all obtained the expected phenomenon as just explained. One observed that:

"As the value of \( h \) becomes very very small, the error becomes large again (in absolute value). When \( h \) is larger, the error is positive. When \( h \) is very small, the error is negative."

One student had even been able to find an \( h \) value that obtained the smallest error one could find. There was another student who commented that:

"I tried (\( h \) equal to) 1 and it complained with a beep sound."

This beeping sound signal was because the author felt that it would not make much sense if students should try any value that was one or greater. Therefore the program was so written as to bar all unreasonable inputs. This sort of error trapping routine is always recommended. A maximum number of trial inputs from a student had also been set to prevent them from having too many tries.
The second example providing hands-on participation for students was to sum $1/n$ $n$ times. The computer initially generated three random numbers between 10 and 50 for the n's and evaluated the sum of the reciprocals to see how far they were from unity. The differences were printed. The students were then invited to try some values for themselves. A message was given to them which stated:

"Try some numbers yourself. Remember, I am very slow, so don't give me any value larger than 50."

Actually, a larger number can produce a better rounding error effect, but it takes a longer time for calculation. Since the APPLE computer is already slow in itself, and there were quite large computations involved in this example, such as the generation of random numbers, the evaluation of the reciprocals of the numbers, and the summations of the fractions, in addition to the use of the authoring language, the slowness of the computer was magnified. Therefore, in order for the students not to have to wait for too long, a value less than or equal to 50 was specified. A subroutine was then added to check that all input values were within the range of 1 to 50. This range should suffice for the demonstration of rounding and truncation errors.

Question four then asked which numbers they observed had
error zero. As it turned out, most students commented that:

"None of them has error zero."

There was one student who stated that:

"It has error zero only when we let $n$ lie between 1 and 9."

Although careful program planning was made in this part in order to minimize the run time, the freeze time was still noticeable and was commented upon by some of the students in the later section of this questionnaire.

Question 5 asked the students to factorize a given degree 4 polynomial using Horner's Rule which was introduced in the lesson. This involves the understanding of the lesson content. As a result, all of the students were able to give the correct factorization.

The five questions in this section were concentrated on knowledge of the subject content. The students were ultimately doing satisfactorily. Although it is too early to draw a concrete conclusion on the accomplishment of CAL, it is still worthwhile to note the students' excellent performance. Other comments and feedbacks are to be considered and will be discussed next.
6.3 Questions on Student's Particulars

Section B of the questionnaire consisted of five questions about the student's particulars on using this package. The questions were:

1. How long did you take to finish the package?
2. Can you type? If you can, how fast?
3. What computer experience do you have?
4. Do you have a personal computer?
5. Have you taken any similar package like this one before? If so, what package?

These questions were trying to find out some background of the students in the computer related aspects and the time taken to use this package. In Question 1, it was found that the time given by the 83 students to finish the package varied from 10 minutes to 60 minutes. The mean time was 28.54 minutes and the standard deviation was 8.59 minutes. The mode was found to be 25 minutes and the median was 26 minutes. Observation of the Normal Probability Plot of the result showed that the time needed for students to finish the package was not exactly normally distributed. Other analysis was used to test the normality; the Goodness of Fit Test rejected the null hypothesis that the time distribution using the package was normal within the 95% confidence interval. Since the distribution showed positive skewness which implied skewness to the right, it suggested that there
were more students who finished the package in less than the mean time than there were who took longer than the mean time. Table 6.1 gives the Stem-and-Leaf Plot, the Boxplot and the Normal Probability Plot of the result from the SAS (Statistical Analysis System) package.

As has been suggested, no single teaching strategy was indicated or how long a CAL package should run. Some students got tired of the computer after 5 minutes while others might work happily for an hour or more. However, many educationalists recommend that a computer lesson should be less than thirty minutes. Heaford [54] gives a graph illustrating that students show a saturation in performance quality if a continuous training session is over 30 minutes. He suspects that during long, highly technical training sessions, the human psychomotor system is bombarded with too many training stimuli, and it becomes difficult to organize the information into an acceptable form. This causes an interference effect, which confuses the laying down of the appropriate motor pattern in the nervous system, hence reducing one's concentration and performance. However, if there is a change of activity or a rest after some time interval, the deterioration in technique or the unstable performance of the skill is not shown. The results of the part of the questionnaire asking for students' comments and of the questions asking about the length of the package (see the next section) also supported this argument. Therefore, the author believes that a single CAL
Table 6.1  Stem-and-Leaf Plot, Boxplot and N.P.P.

<table>
<thead>
<tr>
<th>Stem Leaf</th>
<th>#</th>
<th>Boxplot</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 000</td>
<td>3</td>
<td>*</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 0</td>
<td>1</td>
<td>*</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 00</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3 55555555</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>3 000000000000000000001223</td>
<td>25</td>
<td>+-----+</td>
</tr>
<tr>
<td>2 5555555555555555555555669</td>
<td>25</td>
<td><em>-----</em></td>
</tr>
<tr>
<td>2 000000000023334</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>1 55</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1 0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Multiply Stem-Leaf by 10**+1

Normal Probability Plot

62.5+

---+---+---+---+---+---+---+---+---+---+
-2   -1    0    +1    +2
lesson should not be run continuously for a duration longer than thirty minutes.

From Question 2 of this section, it was found that only 36 students out of 83, approximately 43%, could not type. Out of the other 57%, more than half claimed that they could type at a speed of between 20 and 35 words per minute while the rest mostly said that they could only type slowly. This showed that the majority of the students would not be handicapped because of unfamiliarity with the randomly arranged keyboard. Any such unfamiliarity would, of course, increase the time taken.

In Question 3, it was asked "What computer experience do you have?". There were 44 students who indicated that they did not have any computer experience at all. Among those students who claimed that they had previous computer experience, three said that they had actual work experience on computers in a summer job, five of them had taken the computer subject in the secondary school, 2 others had taken some commercially run computer short courses, and the rest had included the Polytechnic computer curriculum as their first computer experience. Therefore, actually about all of them, apart from these ten students, can be considered as having had no previous computer training before the use of this CAL package.
Question 4 showed that there were 21 students who had a personal computer of their own. However, the types of computers they had and the purpose for which they were used were not specified.

Question 5 asked if they had used any CAL package before. It showed that 74 students, about 89%, had not. The remaining 9 students who claimed that they had used some kind of CAL package were found to be:

- 3 had tried CAL at some exhibition which was mostly for demonstration purpose
- 2 had tried CAL on the word processing tutorials package
- 2 had tried CAL in the BASIC computer language tutorial package
- 1 had tried CAL in chemistry on the topic oxidation
- 1 had tried CAL in game playing.

Strictly speaking, there were only 5 students who had experience on CAL. The other four students only had an informal encounter. Due to this reason, the students' opinion on the package they used would provide very objective and first hand information without bias. Section C and section D of the questionnaire will reveal their points of views on CAL.
6.4 Feedback Questions

There were ten statements and two questions in this section concentrated on the feedback of the students on using the CAL package. These were:

1. The procedure of using the package is simple.
2. The length of this package is about right.
3. The screens are easy to read.
4. Interaction between user and computer is adequate.
5. The content of this package is well presented.
6. The examples are easy to follow.
7. Accompanying written lesson notes are helpful.
8. Use of a package like this in Numerical Analysis is appropriate.
9. If time allows, I would like to have more computer packages available.
10. I would like to borrow a package and use it at home or office.
11. Please describe briefly anything you particularly like about this package.
12. Please describe briefly anything you particularly dislike about this package.

The ten statements were given as point scale choices. A six-point scale ranging from 6 to 1 corresponding to 'strongly agree' to 'strongly disagree' respectively was given. The students were asked to choose the entry that
they thought was most appropriate to apply to each of the statements. The last two were open-answer questions where the students could freely write their own comments.

The statistics, including the arithmetic mean, the standard deviation and the skewness, of the ten statements' responses are given in Table 6.2. Their ranking according to the arithmetic means in descending order together with the corresponding coefficients of variation are given in Table 6.3. The skewness given is the measurement of symmetry of the distribution with respect to the vertical axis. The sign of this measure of skewness, according to Pearson, indicates the direction, whilst the numerical value indicates the strength. The coefficient of variation is a measure of relative variation which can be used to compare the variability in different sets of data.

Analyzing the data, Statement 1 obtained the highest arithmetic mean, which was 5.59 on the six-point scale. There were 56 students out of 83 who gave the score Point 6 and 23 students who gave Point 5, a total of over 95% for these two points. This means that almost all the students agreed that the procedure of using the package was simple. Statement 2 was "The length of this package is about right". It had a mean of 4.90 and ranked sixth among the ten statements. Inspecting the frequency distribution, there was no student who gave the scale Point 1 (strongly disagreed), only 1 student who gave the scale Point 2 and 3
Table 6.2 Statistical Results in Section C

<table>
<thead>
<tr>
<th>Statement Number</th>
<th>Arithmetic Mean</th>
<th>Standard Deviation</th>
<th>Skewness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.59</td>
<td>0.72</td>
<td>-2.47</td>
</tr>
<tr>
<td>2</td>
<td>4.90</td>
<td>0.84</td>
<td>-0.72</td>
</tr>
<tr>
<td>3</td>
<td>5.18</td>
<td>1.06</td>
<td>-1.19</td>
</tr>
<tr>
<td>4</td>
<td>4.59</td>
<td>1.07</td>
<td>-1.06</td>
</tr>
<tr>
<td>5</td>
<td>5.00</td>
<td>0.80</td>
<td>-0.51</td>
</tr>
<tr>
<td>6</td>
<td>5.33</td>
<td>0.80</td>
<td>-1.25</td>
</tr>
<tr>
<td>7</td>
<td>5.13</td>
<td>0.88</td>
<td>-0.59</td>
</tr>
<tr>
<td>8</td>
<td>4.63</td>
<td>1.04</td>
<td>-0.58</td>
</tr>
<tr>
<td>9</td>
<td>4.73</td>
<td>1.13</td>
<td>-0.61</td>
</tr>
<tr>
<td>10</td>
<td>4.33</td>
<td>1.34</td>
<td>-0.56</td>
</tr>
</tbody>
</table>

Table 6.3 Ranking By Arithmetic Mean

<table>
<thead>
<tr>
<th>Rank</th>
<th>Statement Number</th>
<th>Arithmetic Mean</th>
<th>Coefficient Of Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>5.59</td>
<td>12.81</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>5.33</td>
<td>14.98</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>5.18</td>
<td>20.48</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>5.13</td>
<td>17.15</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5.00</td>
<td>15.93</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>4.90</td>
<td>17.04</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>4.73</td>
<td>23.80</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>4.63</td>
<td>22.57</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>4.59</td>
<td>23.33</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>4.33</td>
<td>31.08</td>
</tr>
</tbody>
</table>
students who gave the scale Point 3. Examining the two open answer questions in this respect, there were not too many remarks on the length of the package. There were only 2 students who commented that the package was too long, but there were also 3 students who commented that it was too short. As will be understood, there is no general measurement, both to educationalists and the students on the absolute precise length of a package. It depends on the pace and the interest of the user. Only a majority measure can be sought. The standard deviation and the coefficient of variation were found to be 0.84 and 17.04 respectively and the statement ranked sixth, which showed that the deviation from the central tendency lay in the smaller range compared with the others which implies that the length of package was very reasonable.

Statements 3 to 6 concerned the presentation of the package. Statement 3, which claimed that "The screens were easy to read", had a mean value of 5.18 and ranked third. This showed that students were very satisfied with the screen setting. In Statement 4, a relatively low mean in the "Interaction between user and computer was adequate" was observed. The mean was only 4.59 but the mode and the median were both in the scale Point 5. The relatively large standard deviation suggested there might be some extreme values appearing in the data. By looking in detail at the distribution of the responses, the frequency table had shown that there was one entry given the scale Point 1 and four
entries in the scale Point 2, which were 6% of the total population. Extreme values certainly affected the arithmetic mean. Although most people feel that interaction is very important in a CAL package, there are other people who find that this interrupts the development of ideas. The extent of interaction between user and computer is subjective. On further inspecting the coefficient of variation, it was noticed that it had fallen in the highest score category compared with the others. This might suggest that the standard of interaction was rather diversified among the students.

In statements 5 and 6, "The content of this package was well presented" had a mean value of 5, and "The examples were easy to follow" gained the mean score of 5.33, which was the second highest of all. These indicated that the students were very content with the examples and the layout of the package.

Statements 7 to 10 concerned with the application of the CAL package. In Statement 7, most students thought that "Accompanying written lesson notes are helpful". Although the mean value for this question was 5.13 and ranked fourth, this was the only question that had no response less than the scale Point 3. CAL is neither a copy of the available text book, nor intended to replace the normal printed teaching material. They should be the complement of one another. More detail on the new development concept of
"Computer Enhanced Learning" (CEL), as introduced by Bajpai [50], will be discussed in the next chapter.

The responses on "Use of a package like this in Numerical Analysis is appropriate" had a mean value of 4.63, which was the third lowest score. The fact that students did not give a more positive score on the suitability of using a package in Numerical Analysis may be due to the slowness of the APPLE computer. It is believed that if a faster machine had been used, the score would have been higher.

Students quite agreed that, in Statement 9, "If time allowed, they would like to have more computer packages available". In fact, they showed interest and enthusiasm in using it. However, the last statement "I would like to borrow a package and use it at home or office" had the lowest mean score. This question also had the highest standard deviation and coefficient of variation. This might be due to the fact that not many of them had a personal computer and they were consequently aware that it would not be much help even if they could borrow a package and use it at home.

There were two open-answer questions in this section. As it is difficult to analyze open answer questions, a summary of the students' comments will be given. Question 11 asked the students to describe briefly anything they particularly liked about the CAL package. Most students commented on the
good design and clear presentation of the package. As one student wrote:

"The materials presented are well organized and very easy to understand."

Another student said:

"I think that the content is easy to follow. Examples are enough. It is good because examples can help me in understanding the content."

The next common remarks given by students were about the interactive parts of the package. As one of the students, Cheung, wrote:

"I particularly like the interaction between users and computer. It is adequate and I can go through it according to my capacity and speed of progress."

Yau also wrote that:

"I like the topic about rounding and truncation error because I can input some data to check the error."
There were a few students who commented on the friendliness of the package. Some were even impressed by the sound effects that were produced. One student, Ng, wrote that:

"It seems like talking with the person who wrote the package directly."

Speaking overall, students accepted this new learning approach and they enjoyed using the package. But in Question 12, which looked at the other side of the feedback, and asked the students to describe anything they particularly disliked about this package, the comments were comparatively more diversified than those of the previous question. There were quite a number of students who simply put down "Nothing" and about one fifth of the students mentioned the slowness of the computer. Other comments given by the students were:

- too much detail
- should be more detailed in content
- not enough graphs and sound effects
- not challenging
- cannot ask questions
- screen too small
- cannot be used in other computers
- too long
- too short
Some of these points might be found to be contradictory to each other. This may reflect the different points of view of students. There were comments not directly related to the package itself but to the computer architecture, such as that the screen was too small and the package cannot be used in other machines. As a matter of fact, one of the roadblocks of development of CAL packages is the non-portability and incompatibility between computer systems. Software developed in one machine cannot be used in another. Due to the rapid change in electronic technology, many potential teachers fear that by the time a CAL package is developed and well tested in a computer system, that computer system may be obsolete with the result that the package is completely useless and has to be written all over again. This undoubtedly would discourage a large number of prospective developers.

6.5 Open Answer Question

There was only one question in Section D. It asked students to make comments, in general, on computer aided learning. They might include the advantages, disadvantages, and/or suitability in the Hong Kong environment. Since CAL is a rather new teaching method in Hong Kong, it is expected that not too many students had experience of CAL. In fact, from the results of the previous section in this evaluation, there was only one student who said that he had used CAL in
the school curriculum before. Therefore, their objective and neutral opinions on the first encounter of CAL in academic subjects would give much food for thought.

In their comments, students were able to identify many of the advantages of using CAL in teaching and learning. Some saw that other than being used in traditional teaching, CAL allowed various other modes of presentation to the handicapped, for example, audio for the blind, visual for the deaf, and special control features for other disabilities. If CAL was used in simulating other physical or social systems, it would allow users to raise hypothetical questions and to observe variations of conditions. As to the disadvantages of CAL, some students brought out the point that high level conceptual theories taught without the aid of a teacher were questionable. One student wrote:

"Replacing the human teacher by machine may cause psychological harm."

In fact, perhaps it is the most significant objection to computer assisted learning that very little is known about the long term effects on students learning from a computer. Although impressive results with reading and spelling scores have been demonstrated in short term studies, there are still many unanswered questions about performance and learning over a long period of time.
Concerning the suitability in the Hong Kong environment of using CAL, most students agreed that CAL, although still unpopular, should be used particularly in Hong Kong higher education sectors. As Cheung put it:

"This is a great feature which cannot be accomplished in traditional lecture method."

And Yuen also said that:

"Traditional teaching method in Hong Kong is so called ‘feeding duck’ which is inefficient and stifles innovation to many students. CAL is much more active and interesting. It helps to reduce the student’s pressure on studies."

However, there were students who had reservations on CAL in some of the subjects. Lee commented:

"The students in Hong Kong are mainly Chinese, but the language used in computers is English."

Although Chinese characters can now be displayed in many computers, it is, due to the complicated structure of the Chinese characters, still in the word processing stage and cannot be used in data processing applications.

Some students were worried about the feasibility of
introducing CAL in secondary schools because this may imply a new allocation of resources in terms of manpower, equipment, accommodation and timetabling, etc. The most difficult problem which besets any research into computers in education is finance. Under present conditions, the acquisition of a new skill too often imposes an unduly heavy pressure of work on conscientious teachers. In addition, there is not enough space for schools to house computers in their already fully utilized classrooms. But from the pedagogical point of view, education should not be measured merely by the cost. Other educational values, such as effectiveness on learning and psychological effects should also be considered.

6.6 Summary

Microcomputers have already found their way into homes and commercial firms, as well as into schools. Their numbers are multiplying exponentially. Computing in most Hong Kong secondary schools is still restricted to being a subject on its own. In order to make full utilization of the resources in schools to enhance teaching effectively, computer assisted learning activities across various disciplines, particularly in mathematical subjects, should be sought. In Britain, CAL has been brought into classrooms for many years. The roles of computers in the classrooms have changed the learning activities and experience among
students and teachers. The author, having given a CAL package to a group of 83 year 1 students in the Hong Kong Polytechnic, tried to evaluate and provide information on the effect, effectiveness and successfulness of using the learning package. Guidelines were set to measure some possible variables. They included the identification of the students' weaknesses and the need to provide remedial tuition, and the extent to which each student achieved the objectives using the material. Also, the provision of records of the students' background and using the learning material, and the appropriateness of the learning package were to be measured. A questionnaire, which was divided into four sections, was given to students to evaluate the variables on these aspects after they had used the CAL package.

Section A of the Questionnaire consisted of five questions which were concerned with the content material of the lesson. The first two were straight factual concepts. The third and fourth were questions on the participation of the two examples. The last question involved the understanding of the material and the use of the learned knowledge to factorize a polynomial. These questions were not meant to be post-test problems, rather, they were only treated as students' self-assessment and an indication of the learning progression. Educationalists always emphasize that a three-hour examination or an one-hour test should not be used as the only indicator of students' ability on that
subject, other variables such as students' interest, potential and enthusiasm should also be considered. The result of this section showed that the students were doing extraordinarily well. They demonstrated their competence over the content material on using CAL. Some of them showed more than just the comprehensive knowledge of the topic and explored further findings by themselves.

Section B consisted of five questions which asked the students' particulars related to the use of this package. It was found that the mean time for the 83 students to finish the package was less than 29 minutes and the statistical analysis showed that more than half of the students finished the lesson in less than the mean time. The author believes that this run time for a CAL lesson is most suitable to learners because it has been suggested that the human psychomotor system would become saturated in performance quality if a continuous training session is over 30 minutes and that interest and motivation will not be maintained if saturation is reached.

The time that the students needed to finish the lesson should not be affected by the unfamiliarity of arrangement of the keyboard because more than half of the students claimed that they could type. It was also found that most of the students had not used a computer before and only 29 of them said that they had a personal computer. All of the students, except one, had not used CAL in the school.
curriculum and this was their first time using such a CAL package in an academic subject.

Their feedbacks on the software package were obtained in Section C of the questionnaire. Ten six-point scale statements and two open-answer questions were given. Speaking overall, students showed a very positive attitude to the CAL package in that the highest mean score was 5.59 and the lowest mean score was 4.33 where scale Point 6 corresponded to strongly agree and Point 1 to strongly disagree. The ten statements could be classified into three categories, they were the usage, the presentation and the application of the package. Students agreed that the procedure of using the package was simple but they showed some deviation in opinion on the length of the package. Two students commented, in the later questions, that it was too long and three commented too short. The students generally agreed that the screen was easy to read, the content was well presented and the examples were easy to follow. However, it seemed that they would like more interaction with the computer. Most students thought that accompanying written lesson notes were very helpful. The fact that responses on "Use of a package like this in Numerical Analysis is appropriate" produced a relatively low score, may be due to the slowness of the APPLE computer. It is believed that if a faster machine had been used, the score would have been higher. Students showed that they would like to have more computer packages available but their
opinions on borrowing a package to use at home were very diverse. The reason for this might be due to not too many of them having a personal computer and they were aware that borrowing a package would not be much help.

The two open-answer questions in this section allowed the students to express what they liked and disliked about the package. Most students felt that CAL could help them to understand the subject and arouse their interest, and they enjoyed using it. Other comments were concerned more or less with the architecture of the computer, such as, screen too small, the package could not be used on another computer, etc., which aspects were not in the control of the author.

The last part of the questionnaire was to ask students to make comments, in general, on computer assisted learning. They saw the advantages of using CAL, such as; permit students to learn at their own pace, interactive, can be used in teaching special subjects, etc., however, they were also worried about the psychological effects that would be created in students, and which are still uncertain to educationalists. Concerning the suitability of applying CAL in the Hong Kong environment, they felt that it should be promoted, especially in the higher education sectors, although they realized difficulties would be introduced in reallocation of resources in terms of manpower, equipment and timetabling, and the use of Chinese in computers.
In conclusion, for CAL material to be acceptable, it is essential that it should provide something that other learning systems do not. The advocates of CAL usually put forward an argument based on the ability of the computer to interact, regulate the pace of learners, cope with several students simultaneously, assess and evaluate both students and lessons, and handle large volumes of information. In the context of mathematical subjects, the computer can also provide, via a random number generator, a flexibility which other systems cannot readily make available. The use of CAL was found to be very well received by the year 1 students in the Hong Kong Polytechnic. It succeeded in stimulating positive attitudes about the subject material and learning. Evaluations on students using another CAL package and the effect and effectiveness of CAL will be discussed further in subsequent chapters.
CHAPTER 7

Computer Assisted Learning

7.1 Introduction

The computer was originally seen as a very fast "number crunching" monster. The use of computers to manipulate alphabetic information came much later, and the idea of Computer Assisted Learning (CAL) came much later still. CAL is defined as being the use of the computer in the teaching/learning process in a variety of ways [52] :

- to enhance understanding by simulating a process or experiment so that concepts and interactions are better appreciated
- to help mastery of practical or technical knowledge, sometimes referred to as "drill and practice"
- to provide calculation and analysis facilities so as to remove the burdensome element of much of the tutorial and laboratory work and allow greater time for real learning
- to allow comparative experience to be drawn upon by harnessing the data and analysis capability
of the computer.

There are many names other than CAL which are used in the field of computer education [53], such as, CAI (Computer Assisted Instruction), CBL (Computer Based Learning), CML (Computer Managed Learning), to name a few. Although their definitions do vary and someone may even differentiate between them in minute detail, they all involve similar ideas.

CAL had not been given too much attention by publishers, computer software houses and computer manufacturers as late as the beginning of the 1970s. There were criticisms that contended that CAL amounted to little more than using the computers as an "electronic page turner". Some drill-and-practice CAL was described, even worse, as "electronic flash cards". These pejorative remarks carried all the more impact in view of the extreme cost of early computers, both in hardware and software. To emphasize that computers were costly, one was reminded that they were used to run one program at a time until time sharing, multi-user and multi-tasking concepts were introduced. However, another major element needed to make CAL less expensive, if not economical, was the software to manage the computer and to make it run the user's programs. The early stages of computer language, such as assembler, was too laborious. The later promise of CAL prompted some manufacturers to create specialized languages for CAL authors, such as
TUTOR, NATAL and PILOT. Although the early authoring languages were extremely primitive, they were still much preferred because they saved a great deal of programming time and were more mnemonic. Recently, there is even an authoring system available which is manually driven and guides the CAL authors through the programming process and virtually eliminates the need to know how to program the computer or to know a computer language of any kind. The automation of the CAL authoring process enables developers to focus their activities on instructional logic and the writing of lessons rather than on computer programming.

There are several large computer manufacturers, organizations and institutes who deserve much credit for the development of CAL. Two of the milestones, the PLATO and the MIME projects, are going to be discussed in the following sections.

7.2 The PLATO Project

The PLATO (Programmed Logic for Automatic Teaching Operation) system was the largest and perhaps the most sophisticated computer based system designed for education. This project was funded by the National Science Foundation (NSF) of America and sited in the Computer-based Education Research Laboratory (CERL) at the University of Illinois,
Urbana, Illinois. Control Data Corporation (CDC) also deserved much credit in keeping alive the largest CAL network ever developed. The aims of the PLATO project included the following [54]:

- To develop curricular material for the new medium.
- To develop acceptance by instructor-users and students of a new medium designed for increasing the effectiveness and productivity of the instructional process.

In the PLATO project, there were a variety of instructional applications developed at the college level, which included engineering, physical science, mathematics, foreign languages and nursing. Several of the computer programs were used in college credit courses.

An original contemplation of this project was a complete networking PLATO system that each would be able to support as many as 4,000 individual teaching stations connected to universities and schools both nationally and internationally by telephone lines. Under such a network, it would be possible for a school district to purchase or lease only the number of terminals that it wished to use, hence the expense of installing costly computers and hiring technical personnel in individual sectors would be eliminated. Moreover, duplicated and non-unified CAL software packages
could be avoided.

During its existence, the PLATO I system evolved from a single terminal in 1960 to the PLATO IV system with about 950 terminals in 1977 located at about 140 sites. Each student station interacted with a PLATO program by means of the plasma display panel with an attached keyboard or an Information System Terminal (IST). The major advantage of the plasma panel was that it had inherent memory so that it did not need refreshing and was flicker free. The plasma panel also enabled information from the PLATO program to be either static images from prepared slides, alphanumeric or figures drawn by the computer, or animated motion graphics, or a combination of these. Other devices, such as film projectors, audio and music thesauruses, might be added. The addition of a touch-sensitive panel enabled the students to communicate with the program by touching the screen. Such a capability was particularly useful if the students were asked to rearrange items that appeared in the display and for those who found typing difficult, such as small children.

The PLATO CAL programs could be written in any other available computer language or by the TUTOR authoring language which made it possible for subject-matter experts with no previous computer experience to write their own PLATO lessons without the aid of a programmer. Although the packages did not aim to impose a pedagogical structure on
material, the lessons usually contained minimum material that must be learned by all students as well as supplementary instructional material for students who might find difficulty with a section, and drill and enrichment material for those who wished to try more challenging exercises or desired additional practice. Each student's response sequence, time required and keys that were used were recorded by the computer to permit instructors to track their students' progress through PLATO course material and to be used for later evaluation of the project.

After running the CAL scheme for some time, an evaluation was carried out. The most significant outcome claimed by PLATO was the individualization of education. However, other results of evaluation of PLATO were less categorical in the effectiveness of CAL teaching, as expected. From the data gathered in the computer, they "provide no compelling statistical evidence that PLATO had either a positive or a negative effect on student achievement". From the questionnaires distributed to PLATO users, 27% agreed that "computers are too impersonal for student instruction" and 83% "would not want to have the whole course taught on PLATO [55]. Although the teacher's reactions to the TUTOR authoring language were mostly unfavorable, 88% of them definitely or probably intended to use PLATO again. The evaluation gave the conclusive comment that "PLATO students showed much more favorable attitudes towards computer and computer-assisted instruction than non-PLATO students". The
biggest disappointment of this project came from the fact that it was not cost effective. There were a number of factors which contributed to this. The originally planned 4000 student users per PLATO system had to drop to 1000 due to various reasons, the communication cost for distant users was immense, and the steep drop in hardware prices made small stand alone systems much more attractive propositions. This biggest ever project in CAL was to come to a halt for economic reasons. However, this was not the end of it. Control Data Corporation managed to implement PLATO on microcomputers, which could be linked to the mainframe master system, and opened four PLATO centres in 1980 at Bristol, England, which it claimed would provide plenty of terminal hours and a good study environment. A similar environment was established in NIHE (National Institute for Higher Education) in 1982 at Limerick, Ireland [56].

7.3 The MIME Project

The MIME (Micros In Mathematics Education) project was originally initiated by Professor Bajpai [57], Loughborough University of Technology, England, in the early 1980s. He initially realized that microcomputers would become a useful tool as an aid to mathematics teaching both at university and at school level with their attraction of being relatively inexpensive and their ability to give graphic
displays. By adding some imagination and programming technique, a movement illusion, in effect an "animated blackboard", could then be created on the screen. This facility could aid describing many of the mathematical models and hence improve the teaching of mathematics subjects to all discipline students.

Professor Bajpai discussed his idea of exploiting the use of microcomputers in mathematics teaching with his staff in the Department of Engineering Mathematics. A study was made on what syllabus content should be covered, what sort of material should be produced and how it was to be produced. An investigation on which microcomputer should be employed in the project was also carried out. Based on the fact that microcomputers are very good at producing animated pictures on the screen, some topics of mechanics, such as trajectories of missiles, vibrating systems and collision of bodies, would be very suitable for simulating the physical phenomena by the microcomputer. Therefore, topics of mechanics which appeared in both the "A" level Mathematics and Further Mathematics syllabi of the major examination boards were on the software development list. The hardware, after considering a number of factors [57, P.792-P.793] such as prevalence, price/performance ratio and suitability, was chosen to be the BBC microcomputer model B.

The primary objectives in the construction of the MIME mechanics units included the following:
- To aid understanding of the subject material.
- To add interest where possible by applying practical and relevant examples.
- To be interactive so that ideas can be reinforced and to make learning fun.
- To be user friendly so that there is no need to resort to detailed documentation.
- To be 'idiot-proof' such that accidental or deliberate errors would not 'crash' the software.

The units of CAL packages currently available in Mechanics include:

- Momentum and Impacts
- Projectile Motion
- Linear Motion
- Friction
- Equilibrium
- Vectors
- Relative Motion
- Newton's Laws of Motion
- Work, Energy and Power
- Centres of Gravity
- Circular Motion
- Simple Harmonic Motion
- Angular motion
These packages are all menu driven and written in such a way as to ensure uniformity in format and style of presenting the material, so that once a user got familiarized with the layout of one unit he would be able to use the other units at ease. Each unit had its individual title shown on the appropriate screen display for identification. The main menu provided a PART 0 which gave an introduction for inexperienced users and a PART F which allowed users to exit from the package. The contents in each unit were divided into topics which might further be subdivided into parts. Users could select appropriate topics by either inputting the identified number or the corresponding single letter. A further option would be given if the topic was divided into sub-topics or else the subject material would be activated. In each unit, a user guide and teacher’s notes accompanied the disks. This printed material provided an additional reference source for private study as well as for use in the classroom.

There were some aspects of the units which are worth mentioning and could serve as guidelines and rulings for CAL developers. All units developed could be run on monochrome, as well as colour monitors for full colour support. Sound effects were added to create interest and to aid realism and animation, but could be turned off at the beginning of the unit if they were not desired. This allowed the units to be used in situations where sound effects were not appropriate.
The standardized split of screen display into partitions in all units was used to gain clarity and natural eye movement. For instance, the bottom line was always the information box so that users could look for messages, say, press space bar to continue, in a consistent location. The related information or results evaluated by the computer were either boxed or shown in a structured format. There were certain predefined keys to be used to implement commands for interacting with the computer. For example, the BREAK key could be pressed at any time to obtain the content menu and the ESCAPE key to restart the current part. The use of these keys commands were consistent, mnemonically related to the purpose of the command and minimized the key stroke required so that users were allowed to have full control of the computer at ease but were also able to concentrate on the content of the material without too much distraction.

In about two years's time, the MIME project was able to progress from the scratch idea of using the microcomputer to enhance mathematics teaching to publishing a series of software packages in Mechanics. The first four units were available to the public in January of 1985. Pupils and teachers in local schools had been able to participate in some of the software packages, their reactions to the units being encouraging and favourable.

At present, other than Mechanics, some units in Statistics
are also completed, the names of the units available being:

- Frequency Data
- Probability Distribution
- Bivariate Data
- Sampling
- Inference

The target students in designing the software packages were those who were in the 16 or above age group and were taking mathematics and physics subjects. Although some material might appear to be beyond the level of school students, this was deliberate in the hope that it would give the opportunity to students to get a little insight into a few aspects of the subject beyond their syllabus.

As far as future development of the project is concerned, it is intended to write further series of units to enhance the teaching of mathematics subjects in the sixth-form and at first and second year undergraduate levels. Since the BBC has marketed a new microcomputer model which gives much greater memory capacity, graphic options and runs faster, later units will be based on the use of the new model but compatibility between the two generations will be considered. The availability of the units on other computers is under study. The dynamic instruction set differs very much between machines and conversion is not as straightforward as with a standard high level computer.
language. Moreover, additional support of input media other than the keyboard has been tested successfully in the BBC microcomputer. If it is to be implemented, the software packages may provide a convenient form of remote control for group demonstration or classroom use.

MIME has gained its recognition in Britain. The units are widely used in the nation as well as internationally and we are beginning to see many of the benefits ascribed to it. The ultimate achievement of this project is still to be fulfilled.

7.4 Reasons for CAL

The use of computers in educational institutes has a long history, but the potential of the computer in education is still largely unknown. Many people have tried to define why the computer is being used and what the computer is best at. What might seem relevant is that it rather depends on what one’s expectations are. The administrator’s point of view might not be the same as the teacher’s point of view, even a mathematics teacher may have a very different idea than that of the engineering teacher. However, there is no doubt that computers are here to stay, especially as they are used as a versatile tool in the teaching and learning process of mathematics as a whole.
One of the most significant factors why CAL is being developed so rapidly on microcomputers is that the hardware is becoming both inexpensive and compact without loss of power or capacity as new production techniques and miniaturization are being exploited. Computer technology helps teaching effectively and efficiently as students hear about computers constantly on television and in films, they are "prepared" for computers and even eager to have contact with them. Numerous benefits are to be gained from CAL, for students as well as for teachers, making education a more successful and meaningful experience for both. There are a number of advantages of using microcomputers in education at large, listed in Wellington [53, P.27], whilst some important applications not well discussed but rather relevant to CAL are going to be highlighted.

CAL provides individualized, self-paced learning. Most learning theorists agree that learning is a very individualized process and that the time required differs from student to student. Students have quite different backgrounds and abilities, and they probably differ in all respects from the learning standpoint. These differences are not all known at the present time. Yet, surprisingly, most of the media available for mass education today allow for very little individualization of education. However, having good CAL packages, computers allow individualization responsive to student needs. The inputs and results of the student using the CAL package are
immediately analyzed, and appropriate action can be taken for each different case. This is not to say that CAL programs must have thousands of branches. Indeed, experience indicates that in most cases only a few variants are needed for a specific situation. But these few can make a lot of difference as to how much students will learn.

CAL can facilitate student learning by interaction and immediate feedback from the response. As Piaget suggested, schemata are developed as a result of active involvement. The more active the mental and physical behaviour is, the more effective is the learning process likely to be. A highly interactive CAL requires constant response from students, the activities may include simply answering a "yes" or "no" question, or making same decision, or doing a few calculations. Following the response, an immediate feedback is given. If the answer is correct, a greeting remark can be given and the lesson continues. On the other hand, if the answer is incorrect, the material is repeated but presented in a slightly different fashion until the correct concept is achieved, hence the student is able to correct the mistake in the first instance without any ensuing period of confusion.

There are other advantages of CAL particularly applied to mathematical education using microcomputers. A group of mathematicians has listed a number of computer talents [58],
such as speed of operation, facility to use graphics, randomization, etc., which can be exploited by no other teaching aid. Some of the ideas are being extended and discussed.

Microcomputers can provide colourful, pictorial and dynamic visual displays such as charts, graphs and animated drawings. Graphics are extremely important in the learning process. It has been noticed that typical university level science and mathematics text books contain about one diagram or picture per page. Graphs and drawings of many types are frequently used in classrooms. In CAL packages, dynamics and visualizing animation aid understanding in ways that could not be easily achieved in practice in the sense that motion can be "frozen", analyzed, restarted and traced in whatever sequence wanted. The motions can even be able to superimpose on one another at any discrete time intervals to achieve the sort of effect obtained via high-speed photography. In the MIME project, the units in Projectile Motion, Momentum and Impact traced the motion of moving objects to give a remarkable illustration demonstrating a theoretical mathematical model of a practical experiment.

Another feature the microcomputer can provide over other teaching aids is the randomization. Due to the fast speed of the microprocessor, material presented in CAL need not be in a rigid routine, especially in the mathematics
subjects where algorithms can be well defined. By making use of the random number generator, complete randomness or a specified range of values can be obtained, hence producing a seemingly endless variation of examples and exercises of both a numerical and textual nature. The availability of deviation avoids the problems of students quickly learning the patterns and answers to the exercises and prevents boredom as when using a book.

There are many other advantages that CAL offers such as those given by Doerr [59], for example, as impartiality, patience and motivation. These are all good reasons why CAL can stimulate the learning process, pinpoint student’s weaknesses and upgrade student performance. But there are other potential advantages of using computers in learning that have not yet been envisaged and realized. Nevertheless, students are the motif, learning is their main objective, anything that can increase the process of learning effectively and efficiently, such as CAL, is worthwhile.

7.5 Computer Enhanced Learning (CEL)

Computer Enhanced Learning (CEL) was first introduced by Bajpai [50]. He sees that the roles of microcomputers in the classrooms have now changed. Educators have different
expectations from the CAL packages. CAL is not being used entirely as a manager to replace the teachers as it was in the past, but to support teachers in problem-solving activities in their lessons and to let students investigate, discuss and work on their own with the microcomputers. CAL packages are being used as a tool in teaching, just as is a calculator or a blackboard, and as a catalyst to speed up the learning process. This new trend in the approaches to using CAL can be traced from the recent development of the MIME project [60], the emphasis is shifted from presenting factual knowledge to assisting students in cognitive achievement. The units in statistics are shorter, less "gimmicky", and expect self-motivation on the part of the user rather than aiming to excite his interest.

The aims of using microcomputers in the classroom are interrelated and have to be considered in all dimensions. The multilateral use of the microcomputer serves various purposes, to assist the teaching/learning process as an instrumental tool, to serve as laboratory apparatus, and, at the same time, to provide a means for acquiring knowledge. It is not the author’s intention to attempt to survey the diverse activities in this ever widening area or to look into the distant future, man’s inventive ability being so unpredictable. But CEL definitely has shown much potential in all teaching and learning disciplines. Evolution rather than revolution characterizes development in this field.
7.6 Summary

The great success of microcomputers along with the fast growing computer business have been a major fact in the widespread acceleration of computer assisted learning. With the new technology, microcomputers are capable of providing high resolution colour graphics and are able to facilitate greater expansion of memory and storage, they may also be used as "intelligent terminals" connected to the main computer network hence high speed communication and data transfer is possible. CAL then becomes an effective method of supplementing the teaching process at all levels of education. Computers are used in elementary, junior and high schools as well as in colleges and universities, as a teaching aid in mathematics and in a wide variety of other subjects. No doubt as time and technology advance the versatility of computer-teaching methods will increase. CAL, like books, can be used as a vehicle for the delivery of mass education, students who cannot be reached by a limited supply of teachers can learn from computers acting as a supplementary teacher. Individual help is available to a student who might otherwise be ignored in a classroom. By being able to move privately at their own pace, gifted students are not bound, slow students are not rushed, hard working students are not bored, and shy students are not embarrassed. Moreover, the use of computers can lead to improved student performance in thinking logically, formulating problem solution procedures, and understanding
relationships. The computer is impartial, objective and patient, thus learning is facilitated. CAL can play the vital part in presenting the student with facts he needs in an exciting and interesting way, whilst freeing the teacher to engage in his proper role of coaching and counselling students, and developing his ideas on the facts so presented. Most students are found to respond better to the stimulus of a machine.

However, there are controversial arguments in introducing CAL. The main issues are the availability of software and the implication of a new allocation of resources in terms of finance, manpower and equipment. Speaking of software packages, they are, like books and all other printed matters, not a few men's task or a single term's effort, they take time and need support from all sources. The MIME project is doing a good job. There may be controversy as to what microcomputer system should be used since there are so many different ones available and virtually they are all on their own. It is very true that there is the urge for some form of standardization and compatibility in the somewhat chaotic computer world. And the development of an interpreter to translate the current software from one machine to another is desired. However, these inconsistent varieties should not be seen as a stop sign in the production of CAL packages; there is literature, in one way or the other, in all languages known in the world.
Microelectronic technology has developed at such a rapid pace over recent years that new models of microcomputers come one after another so that the educational system has been caught out by the variety and complexity of the new tools at its disposal. Some people are afraid that the microcomputer they buy today may become an obsolete piece of equipment tomorrow, so they do not want to put their resources into the may be short term investment. However, the view is that equipment bought today will always perform what it can today, tomorrow’s equipment will do more, but not until it arrives. The advantage of starting CAL now is a human benefit in two ways: today’s teacher will learn about the technology that is changing our world and will be in a better position to assess and use the more advanced microcomputers when they come, and today’s crop of students will have some exposure to this aspect of their world rather than passing on out of school without such experience. Therefore, resources would never be wasted, they are just a reallocation and redistribution to meet the changing needs.

There is no doubt that CAL aids understanding by simulating a process or experiment so that concepts and interactions are better appreciated. It also helps to master technical or practical knowledge. Moreover, it provides calculation and analysis facilities so as to remove the burdensome element from much tutorial and laboratory work and allows greater time for real learning. There is one argument that says that a child must learn to communicate with others, if
the student learns wholly alone rather than as a member of a group, due to CAL, psychological harm will be caused. But the point is, CAL will never replace the human teacher, it only augments the teaching in the same sense as do audiovisual aids such as films, tapes and video. Computers will be used increasingly for higher educational purposes in the way Bajpai terms 'Computer Enhanced Learning' (CEL). The teacher, free from the burden of conducting routine drill, will spend more time with individual students, answering their questions, exploring their interests, and inviting them to share his experience.
8.1 Introduction

Knowledge is the building up of information and depends on the ability to learn from past experience. Memory is very important in the learning process. Without memory, humans would not be able to learn and to know what to do. Memory can be categorized as Short-term memory and Long-term memory and involves three stages, as generally agreed by psychologists. These three learning stages are: encoding, which refers to the transformation of sensory information into a form that can be processed by the memory system, storage, which is the transfer of encoded information into memory, and retrieval, which involves locating memorized information when needed, are these three learning stages. Some theorists believe that once an item is learned, it is entered into the Long-term memory and will always be there. Apparently "forgetting" is just a failure of retrieval. There are some factors which help to improve memory, attention and interest being two of these. Attention is influenced by interest, which motivates thinking and which in turn helps in memorizing things. The behaviourists believe that appropriate stimuli during the learning
process, such as interaction and desire response, activate thinking, increase interest and hence improve memory. Psychologists have obtained statistical evidence that people retain knowledge better if they are made to "do" something in their learning process. CAL's interaction is certainly doing. It is interesting to contrast learning from a manual or text book with learning from a computer. Printed materials once written tend to stay the same except for pages of additions and corrections that scarcely make for easy reading. If the text material is replaced by CAL lessons there are many advantages, not the least being that it is easily updated. It also helps people to learn. In fact, it is an increasing trend for computer software instructions and operation updates to be self-documenting by way of built-in CAL lessons. The flexibility of graphics capability in microcomputers marks the advantage of CAL over other learning media. According to the perceptual viewpoint, something that is richly coloured, gorgeous and splendiferous can effectively attract the viewer's attention and help Long-term memory. But unfortunately not too many people recognize that there is a range of abilities and skills involved in understanding graphic presentations. Visual literacy is a neglected area throughout the whole of education. It is often assumed that diagrams and pictures are self-evident, but this is not always true. When a picture represents reality, as in a photograph, there is little trouble. The reader can identify the objects with no effort and the text around the photograph reinforces the
picture. However, if the diagram is to present some hypothetical idea, conceptual information or symbolic references that need interpretation, it may pose translation problems due to differences in culture and experience. Designers should not take their audience for granted. For example, in most western countries, a sign with a fork and a knife implies it is where food is served, but in some oriental countries, a sign with a pair of chopsticks and a bowl is used. This conveys the variety of problems posed in visual literacy and many people seem unaware of the difficulties.

There are four major functions of using pictorial presentations [69], attentional, affective, cognitive and compensatory. These functions are not exclusive and can be interdependent. Attentional illustrations attract or direct the reader to the material. This could happen during the beginning of a CAL package when the student is attracted to the material by seeing an animated logo or when students are asked to refer to a diagram. Affective illustrations are included usually to motivate the students. They enhance enjoyment and affect emotions and attitudes. Cognitive illustrations improve the text comprehension by providing an alternative view of the concepts and additional information to the text thereby extending the students' knowledge. Compensatory illustrations allow for visual presentation to supplement text. This is particularly helpful in those academic disciplines where the academic jargon can obstruct
the reader. The visual message can be understood more easily than the text and can overcome the language barrier.

In producing visual materials, the sequence and order of presentation is as important as the logical sequence of text [70]. It is found that most people read diagrams from left to right and from top to bottom. It would cause understanding difficulties if this sequence of presenting information is altered. As suggested in [72], there are several considerations in using diagrams and graphics in a CAL package if this is to be effective for learning. Firstly, the computer screen size affects the amount of information that can be presented and determines the style, type and size of the illustration. If the text explanation is often on one screen and the diagram on another, an extra burden would be placed on the learners in having to memorize the information to apply to the diagram. So too frequent referring back to the previous program is not recommended. Secondly, the CAL designer may always try to condense the information and alter the diagram to fit the screen. Condensing the information adds a level of difficulty and increases the learning task. A flow chart, for example, is difficult to fit on the screen unless the process is a simple one. Again the unconscious constraint on the designer forces him to fit the design to the size rather than fit the design to the task. The sequence of presentation may be affected. Thirdly, the accuracy and
legibility of diagrams on computers may present problems. Depending on the quality and fineness of the graphics capability, the lines on the screen may be thick and clumsy. Curved and diagonal lines may appear as a series of steps, and line intersections may not be as accurate as desired. It is important to recognize the function and use of diagrams in this case.

The use of colour undoubtedly can motivate CAL users but may actually make the display less legible. The factors inherent in an electronic image include luminosity and non-monochromaticity and it is found that the physical adjustments needed by the eyes to focus colours transmitted on different wavelengths may make the colours appear to jump on the screen. The technical issues are complicated, but anything which takes more effort to read will cause tiredness and loss of concentration. These factors may be subconscious for many people but have been used to explain the reason for easy tiredness in computer users.

The use of computer graphics on CAL packages seems more encouraging as the microcomputers become more powerful and better in graphics capability. The author has developed a second software package, in which the emphasis is on diagrams and illustrations, and this has been used by the students of the Hong Kong Polytechnic. The details of this CAL package will now be discussed.
8.2 The Aim and Content

The aim of the work was to utilize the graphic capability of the microcomputer to develop a CAL software package, to arouse the students' interest in mathematics subjects by applying visual illustrations, and to obtain feedbacks after the students have been using the package. The feedbacks are then analyzed. Results and conclusions are then given on the use of the microcomputer in learning mathematics disciplines.

As was the case in the previous CAL package, the content material had to be decided first. The author, making use of his expertise and teaching experience in Numerical Analysis, still wished to choose this subject area for the second package. Although the recent research areas on Numerical Analysis cover a broader field in mathematics, it is clear that this subject is still as defined by Forsythe [73] in 1959 that:

"Numerical Analysis is the branch of applied mathematics which uses mathematical ideas to devise and evaluate numerical techniques for employing computers to solve problems, and to study their convergence and error."

But in many elementary Numerical Analysis courses, the subject tends to be treated as an exercise in computer
programming and assignments on it assess the students' programming ability rather than his understanding of the techniques involved. The viewpoint of many mathematicians is that it is possible to interest students in Numerical Analysis through the computer, but not by trying to make programming attractive. Coventry Polytechnic in England has developed a microcomputer-based software for use with science and engineering undergraduates [74] the objective of which is to attempt to develop micro usage to enable students to get some sort of feel for the behaviour of the numerical techniques and at the same time to reduce lecture contact time. It is a definite help to let the students understand the numerical algorithm by seeing the results appear right before their eyes. Some students who may make little apparent progress in tutorials might make quite surprising steps forward when actually sitting in front of a terminal. It is of value in teaching the subject if the students' active participation is encouraged.

According to the author's experience in teaching Numerical Analysis, there exists an illusion among students in this field. The students really do tend to believe that numbers are accurately calculated by a computer and that algorithms are absolute. They feel that any treatment of errors and accuracy in ordinary arithmetic processes is a bit of an unnecessary bore. This may be perhaps because they are forced, when not using a computer, to work with fairly simple numbers to reduce the mechanical labour of arithmetic
during closed examples. They do not realize the problems that can arise through the repeated application of a simple process, and how a small inherent error can be magnified by such repetition. Therefore it was decided that a CAL package on numerical integration would be developed and that emphasis would be put on the error incurred.

The contents of the topic were planned to include the Trapezoidal Rule and Simpson's Rule in finding the numerical integral of a given function over a certain domain. These two algorithms apply the same idea of finding the Riemann sum of the integrand to approximate the integral, the only difference is that the Trapezoidal Rule uses trapeziums which imply linear approximation while Simpson's Rule uses parabolas which imply quadratic approximation. These topics involve a repeated simple process of an arithmetic operation and are best illustrated by making use of the computer's graphics capability to demonstrate the relationship between the number of intervals and the resulting approximation error. The lesson was clearly divided into four parts. The first part was the introduction, the second part was the algorithm for the Trapezoidal Rule, the third part that for Simpson's Rule and the fourth part the conclusion.

In the introduction, the reasons why numerical integration is useful were given. The fact is that many indefinite integrals cannot be obtained. As an example, the expression giving the arc length of the simple well known function,
sin x, between 0 and 360 degrees, cannot be solved analytically. But with the help of graphs, it is shown that the solution can be obtained by finding the area under a curve. The distinction between area under the curve and numerical integration was explained. The general idea of numerical integration was then introduced.

The second part of the lesson stated the algorithm for the Trapezoidal Rule and the estimation of the truncation error. A non-trivial function was used as an example. The definite integral of the function in the domain between 0 and 1 was defined. Then the approximations applying the Trapezoidal Rule using 1, 2, 4, 8 up to 128 equal spacing intervals were tabulated to provide a progressive improvement, seen by comparison with the actual value. By considering the truncation error of the algorithm, it was deduced that the error was proportional to the square of the interval used. Therefore the hypothesis "if the interval width is decreased by a factor of two, the error will decrease by a factor of four" was verified by evaluating the ratio of two subsequent errors. The students were then asked to try some interval numbers and let the computer evaluate the error ratio. It was emphasized that the rounding error would begin to dominate the truncation error if the number of intervals used became too large. A summary of the algorithm was given at the end of this section.

The algorithm for Simpson's Rule and the estimation of its
truncation error were introduced in the third part of the lesson. The same function and domain as described in the previous part were used for an example. This would allow the student to compare the superiority in terms of convergence and truncation error of the two algorithms. Simpson's Rule also requires equal spacing and in addition an even number of intervals, therefore the approximations started with 2 intervals, then 4, 8, 16, 32 and finally 64 intervals were used. Since in this case the error is proportional to the fourth power of the interval used, the error should decrease by a factor of sixteen if the interval width was decreased by a factor of two. The students could easily observe that the result of the approximation improved very rapidly without the need of using too many intervals. By further inviting them to try for themselves, it is very obvious that the rounding error begins to dominate over the truncation error suggesting that no improvement could be achieved beyond a certain point. A summary of Simpson's Rule was also given at the end of the section.

The last part of the lesson was a conclusion giving an overview of the two numerical integration algorithms introduced and discussing their derivation and differences.

This content material covered part of the Numerical Analysis syllabus for the Year One Mathematics students. Other parts of the syllabus were taught using the conventional teaching method. A handout, which covered the lesson content in more
detail, was prepared to be incorporated with the CAL package. A copy of this is included in the Appendix F.

8.3 The Approach

An immediate consideration when designing a CAL package is the type of microcomputer to be used. Unlike two or three years previously when the author's first CAL package was in the planning stage, there were at this time so many different types of microcomputers available, each of which had its own format and was not compatible to another so that the personal computer market was in complete chaos. Since the IBM (International Business Machinery) Corporation introduced its personal computer system and got its foot in the market, it gave the lead to the microcomputer industry so that many microcomputer manufacturers emphasized absolute compatibility of their products with the IBM systems. Many software houses developed various commercially used software packages, such as for word processing and spread sheet, as well as powerful compilers and scientific applications that encouraged a wide varieties of computer users. Some unpopular systems which differed from the IBM types were finally ousted by the natural law of supply and demand. From the once doom and gloom microcomputer war it is now becoming clear that some kind of industrial standard is being established. The new microcomputer generation provides for at least 256K bytes of random access memory (RAM), 360K
bytes double sided double density floppy diskette format and a microprocessor clock tick of 4.77M Hertz. In simple terms, these standards allow bigger programs, support a larger mass storage media and run faster compared with the APPLE computer era. The powerful Disk Operating System (DOS) that manages the entire system's resources provides versatile applications on the computer. Although more advanced microcomputer systems have been announced recently, they are all claimed to be compatible with the existing ones except that they are more powerful, run even much faster and have more internal memory.

In the Hong Kong Polytechnic, the Department of Mathematical Studies had about ten Commodore computers which are fully compatible with IBM personal computers for students' use at that time. They all had 640K bytes internal memory, two floppy diskette drives and a hard disk. It is natural that the author should think of developing the CAL package run on these machines. The next thing to consider was whether to use a computer language or to use an authoring system to develop the lesson. In the case of computer languages, there were quite a number of them available, such as C-language, FORTRAN, BASIC etc. Even for just BASIC, there were several versions that can be chosen from, such as the BASICA which is the IBM version BASIC, the GWBASIC which is the Commodore version BASIC, the Turbo BASIC which is a third party version BASIC, etc. Each language had its advantages and disadvantages, some are easier to use, some
run faster, and some provide a more versatile command set. In the case of authoring systems, although they did not have as many choices as the computer languages do, there were still some options.

8.3.1 Authoring System

An authoring system is a computer package which enables teachers to create CAL lessons. It requires no programming knowledge. An authoring system does not require the user to learn a special series of command as, for example, the previously mentioned authoring language SuperPILOT does. The authoring system simply asks the author to respond to prompts and answer questions about the particular lesson or course that he or she wishes to create. When the author has completed the creation of a lesson it is immediately available for review and use without any further processing steps. In fact, even as the author creates a lesson, the author’s view of the lesson is substantially the same as the user’s will be. Two of the authoring systems used in the IBM PC or compatibles systems will now to be discussed.

8.3.1.1 Domino

Domino contains a set of programs that enables CAL developers who do not have special computer training or
knowledge of programming to create software packages in the form of electronic books. Each "book" will be made up of pages which, as in ordinary books, can be arranged in a simple linear chain. Alternatively, through the use of book marks, an author can design a book that guides the reader through it in a predetermined sequence. The sequence followed will depend upon the reader's progress and response to questions. The number of pages per book is only limited by the disk space of the computer. Several books can be stored within the same disk.

The Domino system is extremely versatile and flexible so that it can be put to a variety of applications. The high resolution graphic capability together with the full colour support allow authors to design material that attract the attention and hold the interest of the learner. For example, the drawing of lines, boxes, circles and ellipses of any size and any colour can be easily handled in the graphics mode. A light pen, bit pad or Mouse can also be used with Domino.

Creating CAL packages using Domino is very simple because it is completely menu driven. The authors are guided all the way through a system of commands. Each command may have various levels of subcommands and switches that detail operation of a desired function can be obtained. The material in a page can be listed into a printer, copied or moved to another page. The text can even be centralized on
a page or within a box. It also allows the building and maintaining of an index as any book does. It is certainly a powerful tool for anyone who is interested in developing CAL lessons but does not want to go to the trouble of using a general computer language.

8.3.1.2 MHIAS

MHIAS stands for McGraw-Hill Interactive Authoring System which is an authoring system particularly useful when wishing to involve the learner in practice, follow-up and problem solving exercise types of work. These learning events can be treated as computer based tutorials which present information, provide practice and reinforcement, and give feedbacks. The learner’s performance, number attempted and time used are recorded by the manager of the system and can be recalled and presented. Based on the report of the response, the teacher may analyze and identify the weak points of the learner and hence seek to remedy them.

The basic elements of a package created by MHIAS are the individual screens actually seen by the learners. Each screen created by the author can simply present information which the user is expected to learn, or can be questions, decisions and selections which require the users to react and respond. They are linked together in the way specified by the author. This screen organization is similar to that
of the pages of a book except that rather complex branching from screen to screen based on the response of the learner is allowed. The use of the screen as a basic unit provides both the physical and logical identification and formation of the package. Also, the individual screen allows the selection of the speed at which the material is displayed to the learner - word by word, line by line or a full screen all at once, the format of the lesson display - whole screen, vertical or horizontal half screen layout, and the number of characters per line - 80 characters for normal lettering or 40 characters for larger lettering. The border colour for each screen, as well as the foreground and background colours can also be selected during the creation stage.

The MHIAS is fully menu driven and made up of four major functions - Create and Edit Lessons, Verify Lessons, System Management and Video management.

The "Create and Edit Lessons" menu function is an option to create screen types. Each screen is identified by a unique name supplied by the author. The screen name is the label which provides an entry point for branching or continuation from the previous screen. The screen type can be one of the following five choices, presentation, multiple choice, matching, fill-in-the-blank, and application/simulation. The presentation screen type similar to most authoring systems is used to present text, graphs and charts of the
lesson material. Graphics, such as lines and boxes, are very simple to execute using the keyboard function keys. A very extensive graphics character set which includes mathematical symbols, statistical symbols, Greek letters, etc., is also available. The multiple choice screen type allows the author to set multiple choice questions. The number of alternative answers given for the multiple choice questions can be specified. The correct answer and the control of the branching from each alternative answer can also be stated. The true/false questions can be created using this screen type by simply giving only two branches. The matching screen type is to request learners to give a correct sequence of items to be matched. The author can similarly give the possible alternative answer combinations and branching that complex response ordering can be handled with ease. The fill-in-the-blank screen type allows the author to leave an appropriate blank for the students to answer. The degree of matching of correctness can be either absolutely or approximately right, as specified by the author. The application/simulation screen provides a much more flexible evaluation of answers, for example, it can contain up to 20 answer blanks and the answers can be checked if they are within a given range.

The second major function "Verify Lesson" allows the author to review and edit the complete lesson, to obtain a printed copy of the lesson and to check the accuracy of branching within the lesson.
The third function "System Management" allows the author to configure the system, format, catalogue and copy lesson diskettes and delete lessons and screens from a lesson diskette.

The last function "Video Management" provides a facility to use external video media such as a video tape recorder.

8.3.2 General Computer Language

There are quite a number of general computer languages written for the IBM microcomputers. One of the most common computer languages is BASIC, which was developed in 1964 at Dartmouth College in New Hampshire, U.S.A., as a language for teaching programming. BASIC is an acronym for Beginner's All-purpose Symbolic Instruction Code. Many people found that BASIC is very easy to learn and work, its programs usually take less time to write, and it can do practically almost anything that other high level languages can do. BASIC is available on almost all personal computers. Having many updated versions in more than 20 years since its introduction, BASIC has become a powerful language when working with files, text and graphics in small computers.

There are quite a few BASIC versions available. Some of them are interpreters and some are compilers. In general
terms, an interpreter looks at a computer program one line at a time, checks that for syntax error, resolves to machine readable codes and does whatever is specified if it is error free. If that line needs to be executed many times in a loop, it has to do the same procedure all over again because there is no way to tell from an interpreter which line has been checked. A compiler, in contrast, scans the entire program, resolves all the unknowns once and for all, and executes the instructions. Therefore, the interpretive program runs much slower than the compiled versions but has the advantage of giving better diagnostics in case of error. The author, in view of the heavy demand in computation in Numerical Analysis, is going to investigate the feasibility of using a compiled BASIC computer language in developing the CAL package.

Turbo BASIC is a compiled computer language written by Borland International, Inc. [75]. It is a self-contained programming environment for IBM personal computers and compatibles. It is a combination of editor, fast memory-to-memory compiler, real-time library, and internal linker. Its modern user interface uses windows and pull down menus that enable program development and debugging in an easy way. Unlike interpretive BASIC which requires a line number at the beginning of each statement, Turbo BASIC puts line numbers in as an option or can use meaningful alphanumeric characters as labels in the program. Best of all, the support of accurate and fast floating point
arithmetic gives it an advantage for numeric intensive applications. Furthermore, the enhanced graphic functions and statements make it particularly appealing in developing CAL packages.

8.3.3 The Coding

After detailed investigation of various possible approaches in developing a CAL package and the experience gained in the previous project, the author decided to do the coding in a completely different direction from that of the first package in order to exploit the best part of the computer system. In view of the fact that numerical integration requires very intensive computation, the use of an authoring system would not be appropriate and its drawbacks have been seen in the first CAL evaluation results. Additionally, the author has adequate programming background to cope with the logic of the general computer language so that technical problems should not occur. Therefore, it was decided to write the package in general purpose computer language. Turbo BASIC was chosen because it has a comprehensive graphic command set and is a compiled language which runs noticeably faster. Another change from the first package is that this package is no longer divided into several small programs. One reason for this is that computer memory is no longer the constraint for large programs, one complete program would minimize the overall program length due to
no overheads in passing parameters from one program to the next being needed. Another reason is that it will eliminate the unnecessary access to the diskette hence improve the total run time. However, in the coding stage, the CAL lesson was broken up into four individual program segments according to the content as described previously and was then combined into one program after they were all finished. The major advantages are that they would be easier to expand and to make changes if needed and they could be tested individually in the development stage hence saving debugging time.

The general screen layout of the lesson was planned in the way that the top of the screen was used as an identification line which showed the part the user was currently at. The bottom line would be the message line to instruct the user to response prompt or to interact with the computer. Text material would be shown on the upper half of the screen while graphs and tables would normally be given in the lower half of the screen. All illustrations would be given from left to right and top to bottom as this is the natural eye movement in reading for most people. A similar screen layout format would eliminate the users having to hunt around for information and let them use the package at ease and concentrate on the content.

The first part of the lesson is the introduction. Two windows are created to plot the function $f(x) = \sin x$ and
the equation of the arc length from 0 to 360 degrees to elaborate the insolvability of the problem using an analytical approach. An alternative method of using a Riemann sum to approximate the definite integral is suggested. Another set of graphs demonstrating the difference between numerical integration and the area under a curve is plotted on the screen so that learners can have a physical interpretation of why the sign of the function should be reversed in one of the cases.

The second part of the lesson introduces the Trapezoidal Rule to approximate the definite integral. The function $f(x) = x \exp(-x)$, which is not that trivial but possesses analytical solution for later comparison, is given as an example and plotted. Since this function is used throughout the lesson, it is defined right at the beginning of the program. The value of the function can be obtained by supplying the parameter in a functional call anywhere in the program. A table, used in conjunction with the graph, gives the approximate areas under the function using various numbers of trapeziums. This enables the learner to observe the step by step improvement of the values in successive approximations. The error ratio is further tabulated to elaborate the truncation error estimation. At this time, the learners are invited to input the number of trapeziums and verify the error ratio. The input number will be passed to a subroutine to check for validity before it is used in the evaluation. Also, a maximum number of tries is set to
prevent students having too many 'goes'. A conclusion is given in the instructional type format at the end of this session.

The third part of the lesson gives the idea of the algorithm for Simpson's Rule. The same function and similar screen layout are used so that the user can easily compare the results with those obtained using Trapezoidal Rule. The users are again invited to try some values for themselves. A summary is given at the end of this session.

The last part of the program is the conclusion of this lesson. A comparison of the two algorithms is given. Since the material is mainly in text form, no special graphics effect is needed.

The program was coded in structural form, as far as possible, for easy updating and debugging. Six subroutines were used to handle regularly used procedures and were put at the end of the programs. These subroutines included the checking of input validity, the evaluating of values in the two algorithms, etc. Since Turbo BASIC allows alphanumeric characters as statement labels, all subroutines were given meaningful names, such as INPUTCHECK and PRESSBAR for check input validity and press bar to continue, respectively, for easy identification. Also, Turbo BASIC does not require statement numbers, so these were not used in order to
eliminate confusion. A complete program listing is included in the Appendix F.

8.4 The Use

The CAL package was planned to be used by the first year Higher Diploma students and students in the Degree course in Combined Studies in Mathematics and Science. But before it was actually used, the software had been tested by some students. Some staff also had contributed valuable suggestions. Their general reaction was favourable and they felt that the lesson was interesting, easy to understand and not difficult to use. With the help of previous experience in developing a CAL package, the total time needed from designing the lesson to the completion stage was less than that for the first one.

In the testing stage, it was noticed that the program ran quite fast although extensive graphics had been used. A "DELAY" function had to be put in some parts of the program to slow down the graphic movements in order to demonstrate the effect of progression. In the Commodore microcomputer, it is also possible to add a coprocessor in the hardware to handle all floating point arithmetic in order to speed up the calculation. It is claimed that this can be as much as ten times faster. Since this package involves a lot of computations, it was given a trial run in this type of
computer; as was expected, everything came out in a flash on the screen, and the "DELAY" function did not have any effect at all. Therefore, it was decided that the coprocessor would not be used in the running of the package.

Concerning the actual use of the CAL package, students were given a total of four weeks' time to study the lesson. Since the package was designed to be user friendly and extremely easy to use, and as also there were sufficient microcomputers for their use, the time given to them was more than adequate and they all finished the study on time. The procedure for the Higher Diploma students to use the package was to borrow the disk from the Mathematics Laboratory, put the disk in the diskette drive and switch on the computer. The lesson would then be loaded and start all by itself. When they were finished, they were required to complete a questionnaire, switch off the machine and return the disk. In the case of the Degree Course students, since they were part-time evening students, they were allowed either to study the lesson in the Mathematics Laboratory as were the day-time students or to borrow a disk at home to use if they had a microcomputer.

A sample screen layout of the lesson is given in the Appendix F. The results of the questionnaires will be analyzed and discussed in the next chapter.
CHAPTER 9

Feedback From The Second CAL Lesson

9.1 The Question Paper and the Evaluation

The process by which a student learns is extremely complicated. Whether he is simply learning facts, laws or formulae, changing his values or mastering a skill; the way in which his brain functions is impossible to describe. There are explanations of the learning domains, not in biological terms that happen in the brain, but designed to break the system down into systematic ways so that the learning process can be understood. As suggested by educationalists [78], learning can be divided into three distinct areas, the cognitive domain, the affective domain and the psychomotor domain. The cognitive domain is mainly concerned with what the student will be able to "do", not in a practical sense but in an academic sense, at the end of the lecture or course. The affective domain is concerned with the student’s feelings and attitudes and what he does with the information supplied. The psychomotor domain is concerned with the learning of physical skills, the manipulation of objects or anything that involves "reactions" and neuromuscular coordination. Further, in each domain, there are successive levels through which the
student must pass. However, no matter which domain is being considered, it is found that the learning process takes place in a graduated, step-like procedure, each new level of ability being dependent for its achievement on the mastery of the previous level.

There is an intimate relationship between learning and teaching. The essence of all teaching is communication, not only of knowledge, but of ideas, attitudes and standards. The contribution of visual information to communication is important and is shown by our daily dialogue saying "I see". Visual aids have been used for a very long time in the classroom and have shown their effectiveness. Initially, visual aids were confined to wall sheets, reference pictures or posters. It is only in this mid-century that technical advancement has brought projected pictures, such as slides, and later with sounds, such as films and video, into effective use in teaching. Since the advent of microelectronic technology, the use of microcomputers in presenting pictorial materials opens up another possibility. The graphic capability of the computer can produce variable and alternative illustrations that other teaching aids are incapable of. Anyone who has used a conventional printed form to produce results will know that the essential features of the results emerge only after a few minutes or more of careful scrutiny. A display of a table of numbers will not convey the kind of qualitative information that is needed in mathematics. By contrast, a
computer's graphical display conveys such qualitative information very rapidly and in a form which attracts attention. When graphic presentations are incorporated in the CAL, an even more effective means of communication can be created. The student not only can learn the specified material, but can also use the computer for himself to investigate a problem by varying different parameters. The advantages over class demonstration by teachers are that the student can work at his own pace and follow any line of investigation that occurs to him. As it is believed, knowledge involves the recall of specifics and universals, or the recall of methods and processes, or the recall of a pattern, structure, or setting. For measurement purposes, the recall situation involves little more than to bring to mind the appropriate material. CAL undoubtedly helps the psychological processes of remembering.

The author, after letting the students use a CAL package in which the emphasis is on graphic illustrations, set out a questionnaire to evaluate the variables on some aspects of the students using the CAL package. These aspects include the extent to which each student achieves the objectives, the appropriateness of the instruction style, etc., and were based on the guidelines of objective evaluation as discussed previously. The evaluation was divided into four sections, namely, the Lesson Content Questions, Questions On Student's Particulars, Feedback Questions, and the Open Answer Question Section. In what follows, the responses and
The questionnaire was clearly labelled into four parts, Section A to Section D, according to the four sections of contents. There were 75 Higher Diploma students and 31 Degree Course students, making a total of 106 students involved in using the CAL package. The students were asked to study the lesson first and then complete the questionnaire after they had finished. The lesson disk could be borrowed from the Department's Technician and used in one of the Commodore microcomputers available in the Mathematics Laboratory or, in the case of Degree Course students, used at home if they wanted to. They were given about a month's time to complete the lesson and they all finished the study in time.

9.2 Lesson Content Questions

Section A of the questionnaire consisted of six questions which were all concerned with the lesson content. The questions were:

1. What is the difference between numerical integration and area under a curve?
2. In the Trapezoidal Rule, what is the relation between the error and \( N \), the number of
trapeziums used?

3. In the Trapezoidal Rule, for what \( N \) does the ratio \( \text{Error of } N / \text{Error of } 2N \) begin to diverge? What is that ratio?

4. In Simpson’s Rule, what is the relation between the error and \( N \), the number of intervals used?

5. In Simpson’s Rule, for what \( N \) does the ratio \( \text{Error of } N / \text{Error of } 2N \) begin to diverge? What is that ratio?

6. Why does the ratio not obey the error term for large \( N \)?

These questions were not meant to be post-test problems although they looked as though they were. They were only meant to be treated as a kind of student’s self-assessment questions which were provided as the confirmation of knowledge in the learning process or as the detection of which portions of the learning sequence needed additional remedial options. The questions also give the opportunity to students to get a little insight into a few aspects of the subject beyond factual concepts. These questions would not be graded in the sense that they would affect the student’s final examination score of the Numerical Analysis I result. The students could find most of the solutions in the complementary reading material and they could refer to it when they felt this to be necessary. Out of the six questions the first question involves the basic concepts of numerical integration and area under a curve. In the
introduction of the CAL lesson, appropriate graphs had demonstrated that numerical integration is the exact analogy of the definite integral, hence the sign of the functional value has to be preserved, and that the area under a curve implies the total portions covered by the function, therefore only the magnitude of the function has to be considered. In answering this question, all the students were able to differentiate the difference between the two concepts. Some were also able to identify that:

"If the function does not pass through the x-axis within the given domain, the absolute values of the numerical integration and the area under the curve are the same."

Questions 2 and 3 applied to the Trapezoidal Rule. The function $f(x) = x \exp(-x)$ in the domain between 0 and 1 was given as an example. A table showed, step by step and incorporated with the graph, the approximated values of the areas when using 2, 4, 8, 16, 32, 64 and 128 trapeziums. It can be observed that the approximations got closer and closer to the actual value, except the last one, as the number of trapeziums used was increased. Then the absolute error with reference to the approximations and the actual value and the ratios of the Error of $N$ trapeziums with respect to the error of $2N$ trapeziums were computed and listed in the table. The error ratios obtained should be found to be near 4. The package then invited the students
to try some of their own $N$ values.

Question 2 in the questionnaire asked the students what is the relationship between the error and $N$. In their responses, both the Higher Diploma students and the Degree students were able to observe the phenomenon that if $N$, the number of trapeziums used, is increased by a factor of 2, the error decreased by a factor of 4. Many students even used the mathematics terminology and wrote:

"The error is found to be inversely proportional to the square of $N".

There were a few students who further elaborated the error formula of the algorithm. One student gave that:

"In Trapezoidal Rule, \(\text{Error} = (b - a)h^2M/12\), hence error is proportional to \(h^2\). Therefore, if $N$ increases by 2, then error decreases by 4."

In Question 3, it was asked "For what $N$ does the ratio Error of $N$ / Error of $2N$ begin to diverge? What is that ratio?". To answer this question, it required the students to try different values of $N$. Since the exact ratios used would not obtain the precise integer 4 due to rounding errors (they would be computed as, say, 3.971 or 4.01, values around 4) there is no clear cut point of where the
value began to diverge and it is subject to personal judgement. Therefore, about half of the students noted that:

"For \( N = 224 \), the ratio begins to diverge.
That ratio is \( 3.82758 \)."

And the other half gave that:

"For \( N = 288 \), the ratio Error of \( N \) with respect to Error of \( 2N \) begins to diverge.
That ratio is \( 3.75 \)."

The value of \( N \) that was given is not important in this case, it is the hands-on experience and the exploration of the solution by using CAL that matters. The goal of this lesson is to let them investigate and verify by themselves that in theory the truncation error should keep on improving if the interval width is decreased, but in practice, the rounding error would eventually begin to take over and no further improvement is obtained. This type of experimentation and exploration of alternative solutions given to students definitely cannot be achieved through printed forms of learning media.

Questions 4 and 5 asked the same questions as Question 2 and Question 3 respectively except that the latter questions applied to the algorithm for Simpson's Rule. Due to the
greater superiority of Simpson's Rule over the Trapezoidal Rule, the relationship between the error and \( N \) is that if \( N \) is increased by a factor of 2, the error would decrease by a factor of 16. All the students had stated this fact in answering Question 4. As one student wrote:

"The error is directly proportional to \( h^4 \), i.e., inversely proportional to \( N^4 \). As \( N \) increased by 2, the error decreased by 16. But rounding error dominates quickly."

Students were also invited to try some of their own values. Question 5 then asked for what \( N \) does the ratio of error begin to diverge from the value 16. As mentioned above, the degree of divergence is subject to personal judgement. Fortunately, unlike the Trapezoidal Rule, the rate of convergence in Simpson's Rule is so fast that the rounding error dominates very rapidly. A rather clear cut point of where divergence commenced was obtained. Hence most students gave the response to this question that:

"For \( N = 24 \), the ratio \( \frac{\text{Error of } N}{\text{Error of } 2N} \) begins to diverge. That ratio is 10."

However, a few students felt that:

For \( N = 8 \) the ratio which equals 17.16 begins to diverge."
In the program, there is a subroutine to check the validity of all the student's input, which includes range check and numeric check. Also a maximum number of trial inputs from a student had been set to prevent them from having too many tries in the two examples.

Question 6 was concerned with the basic concept similar to Question 1. It asked "Why does the error not obey the error term for large $N$?". All the students had stated the reason that:

"Because rounding error begins to dominate the truncation error as $N$ increases."

One student further commented that:

"The error may also be related to the significant figures of the computer."

In fact, the Turbo BASIC uses sixteen significant digits in real number manipulation which should provide sufficient accuracy in many cases.

The six questions in this section were all concentrated on knowledge of the subject content. Although their performance cannot be interpreted in the same way as results of a one-hour test or examination, both the Higher Diploma and the Degree Course students were doing more than
satisfactory. Most educationalists agree that potential knowledge of a subject cannot merely be measured by tests and examinations, other factors, such as student’s interest and enthusiasm for the subject have to be considered also.

9.3 Questions on Student’s Particulars

Section B of the questionnaire consisted of five questions about the student’s particulars on using this package. The questions were:

1. How long did you take to finish the package?
2. Can you type? If you can, how fast?
3. What computer experience do you have?
4. Do you have a personal computer?
5. Have you taken any similar package like this one before? If so, what package?

These questions were trying to find out some background of the students in the computer related aspects and the time taken to use this package. Since there were two disciplines of students using this package, the analysis of results will be discussed separately for the Higher Diploma students and Degree Course students. However, results and comments comparing and combining the two groups of students will be given in some of the questions.
In Question 1, for the Higher Diploma students, it was found that the time needed by the 75 students to finish the package varied from 10 minutes to 45 minutes. The mean time was 21.47 minutes and the standard deviation was 6.54 minutes. For the Degree Course students, the time taken by the 31 students to finish the package varied from 10 minutes to 60 minutes. The mean time was 24.84 minutes and the standard deviation was 12.08 minutes. This means that the Degree Course students spent a longer average time on the lesson and deviated more from their time than did the Higher Diploma students. Their modes and medians were the same and were found to be 20 minutes. Analyzing the two groups as a whole, the mean time was 22.45 and the standard deviation was 8.61 minutes. Observation of the Normal Probability Plots of the results showed that the times needed for both groups of students to finish the package were not exactly normally distributed. Other analysis was used to test the normality; the Goodness of Fit Test rejected the null hypothesis that the time distribution using the package was normal within the 95% confidence interval. Since both distributions showed positive skewness which implied skewness to the right, it suggested that there were more students who finished the package in less than the mean times than there were who took longer than the mean times. Using the SAS (Statistical Analysis System) package, Table 9.1 gives the Stem-and-Leaf Plot, the Boxplot and the Normal Probability Plot for the Higher Diploma students, Table 9.2 for the Degree Course students and Table 9.3 for the two
groups of students combined. Further analyzing the result, it was observed that the coefficients of variation were found to be 30.48 for the Higher Diploma students and 48.61 for the Degree Course students. The coefficient of variation is a measure of relative variation which can be used to compare the variability in two sets of data. Hence these figures confirmed that a wider spread of time was required in studying the package for the latter group. The ultimate reason for the diversification of the two student groups was not known, but the results of the part of the questionnaire asking for the students' comments and of the questions asking about the length of the package (see the next section) might provide some explanation.

From Question 2 of this section, it was found that 45 Higher Diploma students out of 75, exactly 60%, could not type, but only less than 26% of the Degree Course students, 8 out of 31, could not type. For those who could type, more than half denoted that they could type at a speed of between 20 and 35 words per minute, but some claimed that they could even type more than 50 words per minute. As this package does not require any alphabetic inputs and it is the author's belief that unnecessary typing by the users should be avoided in order to eliminate the frustration of constant incorrect inputs for those unfamiliar with the randomly arranged keyboard, the knowledge of typing or not should not affect the use of the package.
Table 9.1  HD Student Stem-and-Leaf Plot, Boxplot and N.P.P.

<table>
<thead>
<tr>
<th>Stem</th>
<th>Leaf</th>
<th>#</th>
<th>Boxplot</th>
</tr>
</thead>
<tbody>
<tr>
<td>44</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>0000000000</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>11111111111</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>000000000000000000000000000000001</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0000</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>11</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>11111111111</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>00</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0000</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Normal Probability Plot

-2 -1 0 +1 +2
Table 9.2  Degree Stem-and-Leaf Plot, Boxplot and N.P.P.

<table>
<thead>
<tr>
<th>Stem</th>
<th>Leaf</th>
<th>#</th>
<th>Boxplot</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>00</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 0</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3 55</td>
<td></td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
| 3 0000000 | 7 |       | +------+
| 2    |      |       |         |
| 2 00000000000 | 11 |       | *-----* |
| 1 555555 |      | 6     | +------+
| 1 00  |      | 2     |         |
| 0    |      |       |         |

Multiply Stem-Leaf by 10**+1

Normal Probability Plot

62.5+  
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |
|       |

7.5+  
|       |
|       |
|       |
|       |

-2 -1 0 +1 +2
Table 9.3  Overall Stem-and-Leaf Plot, Boxplot and N.P.P.

<table>
<thead>
<tr>
<th>Stem</th>
<th>Leaf</th>
<th>#</th>
<th>Boxplot</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>00</td>
<td>2</td>
<td>*</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>00</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>55</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0000000000000000000000000003</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5555555555558</td>
<td>13</td>
<td>+-------+</td>
</tr>
<tr>
<td>2</td>
<td>000000000000000000000000000013</td>
<td>37</td>
<td><em>-+---</em></td>
</tr>
<tr>
<td>1</td>
<td>555555555555555555555555578888</td>
<td>24</td>
<td>+-------+</td>
</tr>
<tr>
<td>1</td>
<td>0000022</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Multiply Stem-Leaf by 10**+1

Normal Probability Plot

62.5+  ****

<p>| |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
</table>
|    *
|    **  ++++
|    ++++
|    ********
|    +++++
|    ********
|    ********
|    ********
|    ********
| 7.5+  +++++

-2   -1   0   +1   +2
In Question 3, it was asked "What computer experience do you have?". In fact, all of the students should have been exposed to at least one formal computer course during the term, but it was just to the author's interest to find out whether they had any training in computers other than in the Hong Kong Polytechnics. As it turned out, there were two Higher Diploma students who said that they had taken computer subjects in their Secondary School while two Degree Course students had actual work experience on computers.

Question 4 showed that, unexpectedly, there were more than half of the students who said that they had a personal computer of their own. In exact figures, there were 45 Higher Diploma students and 19 Degree Course students and of these 60% and 61% respectively had microcomputers. Some had specifically indicated that they owned an IBM compatible computer. These figures would definitely affect the result showing the students' willingness to use CAL at home. Further results will be discussed in the following section.

Question 5 asked if they had used any CAL package before. It showed that 68 Higher Diploma students, about 91%, had not. The remaining 7 students, who claimed that they had used some kind of CAL packages, all said that the experience came from the tutorial programs of commercially available software packages, such as DBase III, Symphony, etc. This agreed with the author's previous assertion that built-in CAL lessons for computer software were gradually being
adopted. As to the Degree Course students, since they had been introduced to the concept of CAL in their first computer course, their positive responses on this question were more prominent, as expected.

9.4 Feedback Questions

There were ten statements and three questions in this section concentrating on the feedback of the students on using the CAL package. These were:

1. The procedure for using the package is simple.
2. The length of this package is about right.
3. The screens are easy to read.
4. Interaction between user and computer is adequate.
5. The content of this package is well presented.
6. The examples are easy to follow.
7. Accompanying written lesson notes are helpful.
8. A CAL package like this in Numerical Analysis is appropriate.
9. If time allows, I would like to have more computer packages available.
10. I would like to borrow a package and use it at home or in the office.
11. Please describe briefly anything you particularly like about this package.
12. Please describe briefly anything you particularly dislike about this package.

13. If another package were to be used, what topic in Numerical Analysis do you think should be used?

The ten statements were given as point scale choices. A six-point scale ranging from 6 to 1 corresponding to 'strongly agree' to 'strongly disagree' respectively was given. The students were asked to choose the entry that they thought was most appropriate to apply to each of the statements. The last three were open-answer questions where the students could freely give their own comments.

The statistics, including the arithmetic mean, the standard deviation and the skewness, of the ten statements' responses are given in Table 9.4 for the Higher Diploma students and Table 9.6 for the Degree Course students. Their ranking according to the arithmetic means in descending order together with the corresponding coefficients of variation are given in Tables 9.5 and 9.7 respectively for the two groups. The skewness given is for the measurement of symmetry of the distribution with respect to the vertical axis. The sign of this measure of skewness, according to Pearson, indicates the direction, whilst the numerical value indicates the strength. The standard deviation, or as it is sometimes referred to, the root mean square deviation is the measurement of deviation of the data from the mean.
### Table 9.4 HD Student Statistical Results in Section C

<table>
<thead>
<tr>
<th>Statement Number</th>
<th>Arithmetic Mean</th>
<th>Standard Deviation</th>
<th>Skewness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.39</td>
<td>0.73</td>
<td>-0.76</td>
</tr>
<tr>
<td>2</td>
<td>4.56</td>
<td>1.06</td>
<td>-0.30</td>
</tr>
<tr>
<td>3</td>
<td>5.08</td>
<td>1.05</td>
<td>-0.96</td>
</tr>
<tr>
<td>4</td>
<td>4.17</td>
<td>1.14</td>
<td>-0.68</td>
</tr>
<tr>
<td>5</td>
<td>4.68</td>
<td>0.96</td>
<td>-1.09</td>
</tr>
<tr>
<td>6</td>
<td>4.96</td>
<td>0.96</td>
<td>-1.03</td>
</tr>
<tr>
<td>7</td>
<td>4.79*</td>
<td>0.96</td>
<td>-0.67</td>
</tr>
<tr>
<td>8</td>
<td>4.43</td>
<td>1.03</td>
<td>-0.72</td>
</tr>
<tr>
<td>9</td>
<td>4.79*</td>
<td>1.06</td>
<td>-0.76</td>
</tr>
<tr>
<td>10</td>
<td>4.83</td>
<td>1.13</td>
<td>-0.91</td>
</tr>
</tbody>
</table>

### Table 9.5 HD Student Ranking By Arithmetic Mean

<table>
<thead>
<tr>
<th>Rank</th>
<th>Statement Number</th>
<th>Arithmetic Mean</th>
<th>Coefficient Of Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>5.39</td>
<td>13.61</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>5.08</td>
<td>20.66</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>4.96</td>
<td>19.45</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>4.83</td>
<td>23.44</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>4.79*</td>
<td>20.11</td>
</tr>
<tr>
<td>6</td>
<td>9</td>
<td>4.79*</td>
<td>22.07</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>4.68</td>
<td>20.53</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>4.56</td>
<td>23.15</td>
</tr>
<tr>
<td>9</td>
<td>8</td>
<td>4.43</td>
<td>23.24</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>4.17</td>
<td>27.40</td>
</tr>
</tbody>
</table>

* Same mean
Table 9.6  Degree Statistical Results in Section C

<table>
<thead>
<tr>
<th>Statement Number</th>
<th>Arithmetic Mean</th>
<th>Standard Deviation</th>
<th>Skewness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.58</td>
<td>0.50</td>
<td>-0.34</td>
</tr>
<tr>
<td>2</td>
<td>5.23</td>
<td>0.80</td>
<td>-0.86</td>
</tr>
<tr>
<td>3</td>
<td>5.26</td>
<td>0.73</td>
<td>-0.45</td>
</tr>
<tr>
<td>4</td>
<td>4.23</td>
<td>1.02</td>
<td>-0.29</td>
</tr>
<tr>
<td>5</td>
<td>4.65</td>
<td>0.84</td>
<td>-0.67</td>
</tr>
<tr>
<td>6</td>
<td>4.84*</td>
<td>0.82</td>
<td>-0.07</td>
</tr>
<tr>
<td>7</td>
<td>4.84*</td>
<td>1.04</td>
<td>-0.81</td>
</tr>
<tr>
<td>8</td>
<td>4.52</td>
<td>0.93</td>
<td>-0.72</td>
</tr>
<tr>
<td>9</td>
<td>4.58</td>
<td>1.12</td>
<td>-1.06</td>
</tr>
<tr>
<td>10</td>
<td>4.97</td>
<td>0.80</td>
<td>-0.37</td>
</tr>
</tbody>
</table>

Table 9.7  Degree Ranking By Arithmetic Mean

<table>
<thead>
<tr>
<th>Rank</th>
<th>Statement Number</th>
<th>Arithmetic Mean</th>
<th>Coefficient Of Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>5.58</td>
<td>8.99</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>5.26</td>
<td>13.86</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>5.23</td>
<td>15.40</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>4.97</td>
<td>16.01</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>4.84*</td>
<td>16.96</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>4.84*</td>
<td>21.41</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>4.65</td>
<td>18.05</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>4.58</td>
<td>24.42</td>
</tr>
<tr>
<td>9</td>
<td>8</td>
<td>4.52</td>
<td>20.51</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>4.23</td>
<td>24.22</td>
</tr>
</tbody>
</table>

* Same mean
Comparing the tables, it is interesting to point out that there were some similarities in the results of the two sets of questionnaire. In the ten point-scale statements, there were seven statements which had exactly the same ranking in the two groups of students. The reasons for the three unmatched statements are explainable. The overall means for the ten statements only differed by 0.1, the Higher Diploma students returning a figure of 4.77 while that of the Degree Course students was 4.87. The standard deviations were 1.06 and 0.94 respectively. Detail analysis on individual statements will now be discussed.

Statement 1 ranked first among the two student groups. The Higher Diploma students returned an arithmetic mean of 5.39 on the six-point scale. The Degree Course students even gave a score of 5.58 -- out of the 31 students, 18 gave the scale Point 6 (strongly agree) and the rest gave the scale Point 5. This showed that almost all the students agreed that the procedure for using the package was very simple. In fact, this CAL package made use of the PC DOS (Disk Operating System used on IBM personal computers) AUTOEXEC BATCH command so that the lesson can either be started all by itself after doing the initialization without any intervention on the part of the user or be executed directly under user control. This type of flexibility is recommended because the IBM microcomputers offer a variety of hardware configurations so that a package should be able to cater for both novice users and sophisticated users.
Statement 2 was "The length of this package is about right". It was one of the unmatched statements in ranking and had the most diversified result in the two student groups. For the Higher Diploma students it had a mean of 4.56 and ranked eighth among the ten statements while for the Degree Course students it had a mean score of 5.23 and ranked third. Inspecting the frequency distribution, there was no student who gave the scale Point 1 (Strongly disagree) and only 1 student who gave the scale Point 2 in the former group but in the latter student group, almost 84% of them shared the scale Points 5 and 6. Examining the open answer questions in this respect, there was no student who commented on the length of the package being too short or too long. There was only one student who said that the length of the package was about right. This result further confirms the author's belief that there is no absolute measurement on the right length of a package (see Chapter 6). It is very subjective and depends on the pace and the interest of the user. Only a majority measurement can be sought.

Statements 3 and 6 concerned the presentation of the package. Statement 3, which claimed that "The screens are easy to read", was generally agreed by the students and ranked second in both groups. The mean scores were 5.08 and 5.26 respectively for Higher Diploma and Degree Course students. This showed that the students were very satisfied with the screen setting. Good screen format arouses the
interest of the students and lets them concentrate on the content material. Statement 4, "Interaction between user and computer is adequate", disappointingly obtained the lowest mean scores and ranked tenth and had almost the highest Coefficients of Variation. The means were 4.17 for the Higher Diploma students and 4.23 for the Degree Course students. There were students who commented on this respect in the open answer questions showing that, although they had full control of the pace of the lesson, they still wished to have more "conversation" between themselves and the computer. However, the low ranking for this statement did not indicate that the package lacked interaction between the student and the lesson. The reason for this comparatively low mean score may be due to the author's intention not to include too much "gimmickry" but to design a package based on Computer Enhanced Learning (CEL) objectives, which are to support teaching and to expect self-motivation on the part of the users rather than to excite their interest. The means and medians of the results all lay above Point 4 denoting that the interaction of the package was generally acceptable by the students, but the relatively large standard deviation indicated that extreme responses did exist.

In Statements 5 and 6, "The content of this package is well presented" had a mean value of 4.68, and "The examples are easy to follow" gained a mean score of 4.96 for the Higher Diploma students and were 4.65 and 4.84 respectively for the
Degree Course students. Many students had indicated that the content material was very easy to follow and they particularly liked the way it was presented. The Higher Diploma students in particular enjoyed the hands-on examples which was shown by ranking third according to the mean. As a matter of fact, hands-on experimenting provided in the lesson allowed students to use the computer for themselves to investigate the numerical integration algorithms just as in other science subjects where they can perform their own practicals.

Statements 7 to 10 were concerned with the application of the CAL package. In statement 7, both student groups returned the same rank of fifth that "Accompanying written lesson notes are helpful". CAL is neither a copy of the available text book, nor intended to replace the normal printed teaching material. Computers are regarded as add-on devices. They are technological tools to improve what has already been done with books, films and other media. The response on "A CAL package like this in Numerical Analysis is appropriate" was ranked ninth and had mean values of 4.43 and 4.52 in the two student groups. In fact, many students made reference to the appropriateness of computer graphic illustrations in Numerical Analysis as the suggested topics for another package in answering Question 13. Therefore it is believed that they did think that Numerical Analysis was suitable in CAL and that it was only that their attention
had been shifted to other statements that resulted in a low ranking. In Statement 9, students quite agreed that "If time allowed, they would like to have more computer packages available." In fact, by observing the students in the Mathematics Laboratory, many of them showed interest and enthusiasm in using the package. The last statement, "I would like to borrow a package and use it at home or in the office" ranked fourth. This high ranking certainly revealed the students' interest, as well as the availability of home computers. In the previous section it had been shown that over half of the students indicated that they had a personal computer of their own. Due to the drop in price of the hardware, it will become increasing common for parents to see the computer as just another expense such as a school trip, a musical instrument or a game machine, at least the cost should soon be much the same. From the software point of view, parents will be equally willing to pay for good CAL programs if more development goes into the production of such learning material.

There were three open-answer questions in this section. As it is difficult to analyze open answer questions, a summary of the students' comments will be given. Question 11 asked the students to describe briefly anything they particularly liked about this package. The majority of the students commented on the graphic presentation of the package. As one wrote:
"The graphic presentation is clear and attractive. Together with the text material, the content is very easy to understand."

Another student wrote:

"The content of this package is clearly presented with examples which are easy to follow. Moreover, the summary in the package gives a clear and overall picture of the whole lesson."

Other comments included the hands-on trials, which they agreed could give them a better understanding of the material, and the screen format layout and the slow step by step presentation of the graphs. Speaking overall, students accepted the use of CAL and enjoyed using it.

Question 12, which looked at the other side of the feedback, asked them to describe briefly anything they particularly disliked about the package. There were quite a number of students who simply put down "nothing". There were some students who mentioned the key shortfall of this package:

"It is not easy to go back to the previous screen page."

Since this package was written in general purpose computer
language, it needs some special written entry point routines to handle the flip-flopping of screens. In order not to complicate the package, they were not implemented. From the feedback of the students, it turned out that it was somewhat desired. Other comments given by the students were:

- graphics slow
- limited tries on the examples
- no derivation of formulae
- not colourful
- cannot ask questions
- easy to get tired because of the screen.

Some of these points, such as limited tries on hands-on examples were constraints deliberately added in the program. The slowness in graphic display was due to the inserted DELAY command in order to show the progress of each step. If the program was set to run at normal speed, it would not be easy for users to follow the gradual changes in the graphs. Some other comments, as for example, that which Fung wrote:

"The package does not provide any tests to check whether we understand the information being presented at that moment or not."

which have been discussed in previous chapters and are not
in line with the objectives of this project, will not be pursued further.

Question 13 asked "if another package were to be used, what topic in Numerical Analysis do you think it should be on?". For the Higher Diploma students, many of them gave "Finite Differences and Interpolation", but most of the Degree Course students gave "Roots of Equations" on the ground that graphic presentation of the search for a root might give a better understanding of the rate of convergence. Other popular topics included numerical differentiation, eigenvalue problems and systems of linear equations. However there were also students put down "Any" or "Nil" as their responses.

9.5 Open Answer Question

There was only one question in Section D. It asked students to describe what they thought the advantages and disadvantages would be about using a package like this one in other subjects of their course. In their comments, many students admitted that this was their first experience of CAL. However their attitudes toward using the computer as a teaching and learning aid were favourable and positive. But some of them were a little bit conservative on the use of CAL in every subject of the course. As Leung said:
"I think CAL are only favourable for those subjects which involve a lot of computations, illustrations and graphics. They may not be an advantage for subjects which require a lot of description and rigorous proofs."

Most students were able to identify many of the advantages and disadvantages of using CAL in general. Although their points of view were in the main confined by their limited experience, many of them felt that CAL could only serve as a supplementary aid in teaching and could not replace the role of a teacher. In fact, it has always been emphasized by educationalists that computers are only an add-on device and CAL should be seen as a learning tool which is being integrated into the institutions, curricula and learning philosophies that must be used at the right time, to the right amount, and in the right way.

9.6 Summary

Computers, through their use in a variety of information, communication and manufacturing systems, have the potential of being the most powerful intellectual tools that mankind has ever invented. The computer has also become the symbol for international power and prestige. Many countries have put computer literacy as top priority in their education policy. The use of computers in the education sector
includes all facets. Other than the straightforward computer subject on its own, data processing and fast number crunching machines, they can be used as a teaching and learning aid. There having been many generations of microelectronic technology since its introduction, the microcomputer is no longer a cheap alternative to a mainframe or minicomputer. Microcomputers and other large scale computers are essentially the same in architectural concept, but have the advantages of being small, having graphic capacity and being systems all on their own. Microcomputers give the hands-on capability that, it is believed, cannot be accomplished on a mainframe class of computer. The benefit of using microcomputers in the learning process is that students may gain familiarity with the use of computers while taking part in their exercises. It would cast-off their fear of computers and they can learn a programming language or use other software if they need it. The introduction of CAL has been found successful in schools in many countries but is in the infancy stage in Hong Kong. The author, having introduced the second CAL package to the Higher Diploma students and the Degree Course students in the Hong Kong Polytechnic, would like to sum up the evaluation results obtained. The evaluation was given in the form of a questionnaire which was divided into four sections in order to identify different aspects after they had used the CAL package.

Section A of the questionnaire consisted of six questions
which were concerned with the content material of the lesson. The lesson basically introduced two algorithms in Numerical Analysis. The first question was a straight factual concept. The second and third were questions on the Trapezoidal Rule while the fourth and fifth were on Simpson’s Rule. To answer these questions involved the participation of students in trying these two rules. The last question required the understanding of the hands-on experiment to give conclusive reasoning. These questions were not meant to be post-test problems because most research papers published have shown insignificant outcomes of tests on students using or not using a package. The main reason for this is because it is believed that an examination can only give the score of the students’ performance at that moment, the students’ long term interest and enthusiasm for the subject cannot be measured. Therefore, these questions were only treated as students’ self-assessment and an indication of the learning progression. Nevertheless, the result of this section showed that the students were doing unanimously well. They demonstrated their competence over the content material on using CAL.

Section B consisted of five questions which asked the students’ particulars related to the use of this package. It was found that the mean time taken for the Higher Diploma students to finish the lesson was noticeably less than that of the Degree Course students being 21.47 minutes and 24.84
minutes, respectively. The overall average was 22.45 minutes. This run time was considered to be reasonable as it is the author's belief that a prolonged lesson would reduce the learner's interest and motivation and consequently lose his concentration.

In order to eliminate the frustration of constant incorrect input due to unfamiliarity with the keyboard arrangement, no alphabetic input was required in the lesson. Therefore knowledge of typing or otherwise should not have any effect on the study. Although more than half of the students said that they had their own personal computer, most of them indicated this was their first time using such a CAL package in learning any subject, let alone mathematics.

Section C of the questionnaire reflects their feedbacks on the use of software packages. Ten six-point scale statements and three open-answer questions were given. Speaking overall, the two groups of students showed a very positive attitude to the CAL package in that the highest mean score was 5.58 and the lowest mean score was only 4.17 where scale Point 6 corresponded to strongly agree and Point 1 to strongly disagree. The ten statements could be classified into three categories, they were the usage, the presentation and the application of the package. Students agreed that the procedure of using the package was simple but they showed some deviation in opinion on the length of the package. But no student commented on whether it was too
long or too short. There was only one student who said that the length of the package was right. The students generally agreed that the screens were easy to read, the content was well presented and the examples were easy to follow. However, it seemed that they would like more interaction with the computer. Most students thought that accompanying written lesson notes was very helpful. The fact that responses on "A CAL package like this in Numerical Analysis is appropriate" had a relatively low ranking may be due to their attention having been shifted to other statements. In fact they had emphasized in the open-answer questions that graphic illustrations and hands-on experience helped their understanding of Numerical Analysis. Students further showed that they would like to have more computer packages available and they generally favoured borrowing a package and using it at home. This can be explained by the drop in price of hardware with the result that personal computers are more affordable by, and acceptable to parents.

The three open-answer questions in this section allowed the students to express what they liked and disliked about the package and what other topics in Numerical Analysis they would like to be in CAL. Most students felt that CAL could help them to understand the subject and arouse their interest, and they particularly enjoyed the graphic illustrations and the hand-on examples. Some of the students preferred that the contents of screens could be flip-flopped. Other comments, such as, graphic slow,
limited tries on examples, etc., were deliberately planted by the author to control the running of the lesson. The students' suggestions on other CAL for Numerical Analysis were generally related to the possibility of graphic presentation of that topic, such as "Roots of Equations" and "Interpolation".

The last part of the questionnaire was to ask students to make comments, in general, on computer assisted learning. They saw the advantages of using CAL in most subjects, such advantage being in letting students to learn at their own pace, interaction, graphic capabilities, etc. However, they did feel that subjects which required a lot of wordy descriptions and rigorous proofs might not be a best choice for CAL.

In conclusion, the use of CAL in education will have a profound impact on both the teacher's and students' role, and on the learning situation in general. However, the teacher's role will certainly continue to be central to the education process. But students' comments and opinions help to bridge the generation gaps between themselves and teachers. It is only through extensive research and the building of prototypes that we can effectively explore this media. Just as Bowtell and Haines said at the CTISS Conference at the University of Kent in 1986 and to be published [79]:

"There is certainly a clear case for introducing micro into the teaching of undergraduate mathematics but there is still a lot of work to be done."
10.1 Comparison of the Two Evaluations

The author wishes to give an overall view of the two batches of students after they had been using the CAL packages. In the first CAL package, there were 83 Year 1 Higher Diploma students involved in the evaluation. In the second CAL package, there were 75 Year 1 Higher Diploma students and 31 Year 1 Degree Course students involved in the evaluation. The review will be concentrated on the comparison of students' achievement on the subject material, their background on how the packages were being used and their feedback on various aspects.

Section A of both questionnaires was concerned with the content material of the lessons. The questions included straight factual concepts as well as understanding and participation in hands-on examples. The results showed that they more than just comprehended the topics in the CAL lessons, they had also shown interest and enthusiasm in experimenting by themselves.
Concerning the time needed to finish the CAL lesson, it was found that the mean time for the first package was 28.54 and that for the second was 22.45. Although some difference in the mean time occurred, it is still difficult to compare the lengths of the two lessons. The slowness of the APPLE computer might give some explanation but the times should not be affected by whether a student could type or not because the author avoided alphabetic input as far as possible in the CAL lessons to eliminate frustration of incorrect input. The percentage of students who could type decreased from 57% to 50%.

It was found that there was not much difference in computer experience between the two batches of students but the number of students who had a personal computer showed a significant increase. In the first group, there were only 25% of the students who owned a personal computer while there were over 60% in the second group. This increase, as discussed previously, was definitely due to the reduction in the price of hardware and the acceptance of parents to see the computer as one of the home appliances, such as a microwave oven or a telephone.

The percentages of students who had experienced CAL before they used one of the two packages was very much the same in the two years except that in the second group more students had experience of those tutorial lessons included in the commercially available software packages, such as DBase III
and Symphony.

In the section concerning the students's feedback on the CAL packages, it was found in the ten point-scale statements that the overall mean, standard deviation and the range for the first CAL package were 4.94, 1.04 and 1.27 respectively, while those for the second package were 4.80, 1.03 and 1.25. This showed that the scores given by the students were very homogeneous. The closeness of coefficients of variation, which were 21.14 and 21.42, reaffirmed this. Amongst the two batches of students, most of them had similar views on the two CAL packages. For example, "The procedure of using the package is simple" and "The length of this package is about right" had the same ranking of first and sixth respectively. The other statements in the two questionnaires had similar ranking. The greatest disagreement occurred with Statement 10 which was "I would like to borrow a package and use it at home or in the office". It ranked tenth in the first package but fourth in the second. This deviation is believed to be closely related to the percentage of students having a personal computer as described above. Table 10.1 gives their comparison by statement and Table 10.2 by ranking.

10.2 Summary

Microelectronics enable the computer power previously
Table 10.1  Comparison Results In Section C

<table>
<thead>
<tr>
<th>Statement Number</th>
<th>First CAL Arithmetic Mean</th>
<th>Second CAL Arithmetic Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.59</td>
<td>5.44</td>
</tr>
<tr>
<td>2</td>
<td>4.90</td>
<td>4.75</td>
</tr>
<tr>
<td>3</td>
<td>5.18</td>
<td>5.13</td>
</tr>
<tr>
<td>4</td>
<td>4.59</td>
<td>4.19</td>
</tr>
<tr>
<td>5</td>
<td>5.00</td>
<td>4.67</td>
</tr>
<tr>
<td>6</td>
<td>5.33</td>
<td>4.92</td>
</tr>
<tr>
<td>7</td>
<td>5.13</td>
<td>4.80</td>
</tr>
<tr>
<td>8</td>
<td>4.63</td>
<td>4.45</td>
</tr>
<tr>
<td>9</td>
<td>4.73</td>
<td>4.73</td>
</tr>
<tr>
<td>10</td>
<td>4.33</td>
<td>4.87</td>
</tr>
</tbody>
</table>

Table 10.2  Comparison in Ranking By Arithmetic Mean

<table>
<thead>
<tr>
<th>Ranking</th>
<th>First CAL Statement Number</th>
<th>Second CAL Statement Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>4</td>
</tr>
</tbody>
</table>
available only in large and expensive mainframe computers to be applied to a wider variety of applications. In education, the computer, other than being treated as a study subject on its own, a fast number crunching machine and a data and word processor, should be seen as a learning tool which must be integrated with the institutions, curricula, and learning philosophies of teaching. The use of CAL in microcomputers can supplement classroom instruction and help the student’s learning process. The most important reason for using CAL is that it is uniquely interactive. No other medium, except a human teacher, demands that a student makes frequent responses to ensure understanding. By using the powerful computer, students can often study important contemporary problems without first having to master a great deal of specialized knowledge.

In this research project, the author developed two CAL packages in Numerical Analysis. These packages were used by the students and run on microcomputers. Then feedback from the students and the effect and effectiveness of the use of microcomputers in mathematics in the Hong Kong Higher Education environment were studied. Measurements are based on questionnaires, conversation with students and answering questions on content material. Detailed evaluations are given in previous Chapters. As a whole, the students enjoyed using the packages and they found them motivating and very helpful in the learning process. Their performance on answering the questions on content material is outstanding.
The use of CAL is found to be very well-received by the students in the Hong Kong Polytechnic. The survey also shows very positive feedback on the attitude to applying CAL in learning. The students are particularly impressed by the graphic illustration capability of the microcomputer and they do feel that it really helps in understanding the lesson. But they also point out the difficulties of applying CAL in some other subjects, such as those involving Chinese character display or requiring wordy description and rigorous proofs. Concerning the length of a CAL lesson, the range of the time needed for the students to finish was from 10 minutes to 60 minutes. Although there is no absolute standard of how long a student should study a lesson, it is the author's belief that in any case the average time for a lesson should not be more than thirty minutes or else it would lose the user's concentration and interest and hence reduce his performance.

10.3 Recommendations and Suggestions for Further Research

This research is considered to be a very successful one. There are some suggestions and recommendation for further research. In this study, the cost effect has not been considered. Investment in microelectronic technology is an expensive undertaking for an educational system. The problem of rapid obsolescence of new technology is particularly acute in tertiary institutes. Also, the impact
of the change on staff and student time depends on the extent to which CAL material is applied. This new allocation of resources in terms of manpower, equipment, accommodation and timetabling clearly changes the economic priorities of central government's short-term and long-term planning and could be further investigated.

In the two evaluations, it had shown that for most of the students involved in the use of the CAL packages it was a new experience. It is not certain what would their reactions be when the novelty wears off. Future research could be the optimum use of CAL in teaching. The findings of the best mix of CAL and conventional teaching could be a fruitful area for further study.

Another consideration is the long-term psychological effect. As traditionally understood, the mathematical requirements of a modern society place emphasis more on understanding, less on computational skill. The invention of calculating machines has lessened the burden on humans of pure arithmetic operations. But the study of the long-term effects of students learning from a computer is still uncertain. Although the computer is no longer to be seen as a mysterious construction and it is known that the computer cannot solve everything and can prove nothing in general but is just a piece of everyday equipment, there is still a need for further understanding of the use of CAL in the psychological consequence in the learning process. CAL is
not designed to regiment teachers, nor to replace them, nor to dictate methods, but to support and strengthen their teaching. The personal attention, contact and care of teachers are very valuable for the students and can, in no way, be replaced by the computer.

Final Remarks

The findings of the author's research and the areas which are important for future research are listed below:

(i) Hong Kong Polytechnic students found that CAL motivates and enhances their understanding of the subject material. Software packages for use in other subject areas were also suggested.

(ii) The use of CAL/CEL is recommended in Hong Kong education sector. It helps to reduce the student's pressure on studies in the present so-called "feeding duck" education system.

(iii) The interactive, randomized and graphic capabilities of microcomputers increases the process of learning.

(iv) The length of a CAL lesson should not run for more than thirty minutes for an average student, otherwise the user's concentration and performance is impeded.

(v) The CAL/CEL ideas developed in this thesis need to be disseminated amongst teachers in Hong Kong.

(vi) The promotion of CAL for subjects requiring the use of Chinese characters may encounter some difficulties.

(vii) The optimal mix of CAL/CEL with other teaching methods needs to be further investigated.

(viii) The problem of hardware which directly affects the use of software packages still exists and the best allocation of resource requires in-depth evaluation.
REFERENCES

[Note: Some references and books given below have not been referred to in the text but have been used in the development of the work reported in this thesis.]


[5] Sr. Wong, M., Speech delivered at the seminar on "Computer Studies at the Form Six Level", 1986


Jones, R. "Micros in the Primary Classroom", 1984, Edward Arnold (Publishers) Ltd., England


Maddison, A. "Microcomputers in the Classroom", 1982, Hodder and Stoughton, England


[38] Tinsley, J.D. and Tagg, E.D. "Informatics in Elementary Education", 1984, Elsevier Science Publishers B.V., Netherlands


[40] Hong Kong Government, "Education Commission Report No.2", August 1986, Hong Kong


Beech, G. "Computer Based Learning, Practical Microcomputer Methods", 1983, John Wiley & Sons Ltd., England


Levie W.H. and Lentz R. "Effects of Text Illustration; a Review of Research", 1982, Educational Communication and Technology Journal Vol. 30 No. 4


E.T.U., "Teaching Methodology Papers", 1986, Education Technology Unit, Hong Kong Polytechnic, Hong Kong


APPENDIX A
**TIMETABLE FOR USE OF COMPUTER ROOM**

The Computer Room will be open as follows on normal school days:

- **Mon., Tue., Fri.** 3.40 - 5.00 p.m.
- **Thu.** 3.40 - 4.30 p.m.
- **Sat.**
  - 9.00 - 10.30 a.m. (*)
  - 10.30 - 12.00 a.m. (*)
  - 2.00 - 3.30 p.m.

(*) These sessions are for Computer Studies class students only

**RULES FOR USE OF COMPUTER ROOM**

1. A booking must be made for that practice session before a student is allowed into the computer room.

2. Students taking the 'Computer Studies' course must book for the practice sessions in the next week on or before Friday.

3. Computer Club members must book for practice sessions during Monday lunch time, 1.10 - 1.25 p.m.

4. The computer room will only be open if there are two or more bookings for that session.

5. Students must bring along their student card and show it to the technician in charge of the computer room upon entry.

6. Each student must note down their name, class no., and time of usage each time they practice in the computer room in the log-book next to the computer they are using.

7. Any student arriving later than 15 minutes after the commencement of the practice session will be considered late. Late students must report to the technician on duty (name posted in Computer Room) before starting their practice. Failure to do so would constitute an unauthorized use of the computer room.

8. Students practicing in the computer room are responsible for the safety of all school properties in the computer room, and they must prevent any unauthorized use of the computer room. Thus, when there is no technician in the computer room, no one may be admitted. In case of unauthorized students entering the room, those practicing there, i.e. the authorized users will be punished.

9. When there are only two students left working in the room, anyone wishing to leave must inform the technician on duty first. This is to prevent leaving the computer room vacant when the last one goes to inform the technician.

10. Penalty - anyone late more than once in a month or anyone absent for a practice without valid reason (together with documentation, e.g. parent's or doctor's letter) will not be allowed the use of the computer room for two weeks.

11. Any irregularity in the functioning of the computers should be reported immediately to the technician in charge and a written report of this should be made in the log-book. No one may then use that computer until the irregularity has been put right.
<table>
<thead>
<tr>
<th>DATE</th>
<th>HARDWARE DEVELOPMENT</th>
<th>ACTIVITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nov., '81</td>
<td>The school's first computer - an Apple II microcomputer with one disk drive and a printer was bought.</td>
<td>The Computer Club was set up and was in charge of the display of the computer at the open day of the 138th anniversary of the school. Short sessions on programming were arranged for the Club members. As there was only one computer available, only F.6 students were allowed to join the Club. The computer was used to do the mark and grade calculation as well as the order of merit for the six F.3 classes.</td>
</tr>
<tr>
<td>June, '82</td>
<td>Four more sets of computers were bought - all Radio Shack TRS-80 Model III, three of which have single disk drives attached and the fourth one has a dual disk drive. Another dot matrix printer was also bought.</td>
<td>Formal teaching of Computer Studies as a C.E. level subject was first started. A total of 28 students were admitted. All students from the six F.4 classes can apply for the course and the students were selected on the basis of their order of merit. Lessons take place after school. Practice sessions take place after school and on Saturday mornings. The Computer Club continues to provide opportunities to those not in the computer class to learn more about computers and computing.</td>
</tr>
<tr>
<td>Jan., '83</td>
<td>The new computer room was completed.</td>
<td>Four short sessions were organized by the Teachers' Council to introduce teachers to using the computer to set multiple choice questions and to use it as a word processor. Use of computers to do mark calculation and order of merit score for F.1 - F.3 and the class allocation for F.3.</td>
</tr>
<tr>
<td>May, '83</td>
<td></td>
<td>48 students from this year's F.4 are admitted into the Computer Studies course. (Thus the total no. of students taking this course is 68) The admission policy is the same as for '82. Computer Club membership is extended to include students from F.4 and upwards.</td>
</tr>
<tr>
<td>June, '83</td>
<td>Six new computers were bought - BBC Micro Model B, all connected in a network. A dual drive of total capacity 888K serves the whole network.</td>
<td></td>
</tr>
<tr>
<td>Sept., '83</td>
<td>A new wide carriage printer was bought so as to reduce the waiting time for printing. 5 more BBC Micro Model B were installed into the network. In order that students can have more chances to use the computers, 3 computers were placed in the library. Of these, 2 are new BBC Model B computers each with a cassette recorder and one Radio Shack Model III taken from the computer room.</td>
<td></td>
</tr>
</tbody>
</table>
SYLLABUSES FOR SECONDARY SCHOOLS

SYLLABUS FOR COMPUTER STUDIES (FORMS IV—V)

ONE OF A SERIES OF SYLLABUSES RECOMMENDED BY THE CURRICULUM DEVELOPMENT COMMITTEE HONG KONG 1983

Preface

This syllabus is recommended for use in Hong Kong secondary schools at Forms IV & V level. Besides listing the topics to be taught, it also provides notes on teaching and suggestions for practical activities. The objectives of this course are:

1. To provide opportunity for the study of modern methods of information processing so that students may understand and apply this rapidly growing technology.
2. To acquaint students with the uses and limitations of computers.
3. To develop among students problem solving skills through interaction with computers.
4. To encourage an understanding of the implications of computers in the modern world.
5. To prepare students who wish to go on to further studies in computer science.

This course is largely independent of any other subject taken at HKCEE level. In particular, it contains very little mathematics and there is little overlap between this syllabus and any other syllabus approved by the Education Department.
**COMPUTER STUDIES**

Estimated time allocation

A time allocation of 4 periods per week is recommended for this course. Taking an average school year as 27 teaching weeks, a total of 216 periods is available. An estimate of the number of periods for each section is shown below:

<table>
<thead>
<tr>
<th>Section</th>
<th>Periods</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Evolution of Information Processing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.1 Calculating aids</td>
<td>3</td>
<td>A brief survey only is required to trace the evolution of calculating aids through the ages. This should include:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(i) tally marks</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(ii) counting on fingers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(iii) the abacus</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(iv) logarithms and the slide rule</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(v) mechanical calculators (e.g. the work of Pascal, Leibniz)</td>
</tr>
<tr>
<td>1.2 Mechanical stored program devices</td>
<td>4</td>
<td>Treated briefly. Emphasis should be on the importance of a device which is capable of operating repetitively and automatically as a result of a stored program:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(i) The work of Jacquard and Babbage</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(ii) Modern examples of similar devices (e.g. music box, keymatic washing machine)</td>
</tr>
<tr>
<td>1.3 Mechanical data storage devices</td>
<td>2</td>
<td>This section is intended to emphasise the need for a new form of data storage to handle the rapidly increasing requirements by the end of the 19th century. The work of Hollerith should be discussed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Introduction to the punched card as a means of data storage.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Hollerith cards should not be introduced at this time.</td>
</tr>
<tr>
<td>1.4 The development of electronic computers</td>
<td>2</td>
<td>Early electro-mechanical and electronic computers and the development to the present day. Details descriptions of systems are not required. Emphasis should be on changes in physical size, speed of operation and the impact of transistors leading to the micro-electronic revolution. The rapid growth in the use of electronic devices of all kinds (e.g. calculators, video games etc.) should be stressed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Introductory Computer Concepts</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.1 “Input → process → output” concept</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2.2 The computer</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2.3 Simple programming</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2.4 Hardware of a typical microcomputer system</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3. Flowcharting</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3.1 Flowcharting of non-computer problems</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3.2 Computer algorithms in flowchart form</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>3.3 Conversion of simple flowcharts into BASIC</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>3.4 Importance of flowcharting</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>4. Programming in BASIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.1 Program documentation</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>4.2 Program language</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>4.3 Programming (see below)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Input/Output, Coding and the Storage of Information</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.1 Coding in a non-computer context</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>5.2 Number codes</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

* The remaining time (approx. 86 periods) should be used for students to develop and test their own programs on the school microcomputer. This time should be distributed over the two year period such that students work with the microcomputer on a regular basis.

### 1. EVOLUTION OF INFORMATION PROCESSING

<table>
<thead>
<tr>
<th>Syllabus</th>
<th>Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Calculating aids</td>
<td></td>
<td>A1. Using an abacus</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A2. Constructing and using simple paper slide rules for addition, subtraction and multiplication, division.</td>
</tr>
<tr>
<td>1.2 Mechanical stored program devices</td>
<td></td>
<td>A3. Dismantling a toy music box.</td>
</tr>
<tr>
<td>1.3 Mechanical data storage devices</td>
<td></td>
<td>A4. Close construction of a set of edge-chipped cards storing personal data (e.g. sex, age, height etc.), stored using a knitting needle.</td>
</tr>
<tr>
<td>1.4 The development of electronic computers</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2. INTRODUCTORY COMPUTER CONCEPTS

<table>
<thead>
<tr>
<th>Syllabus Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
</table>
| **2.1 Input → process → output concept** Students should be aware of the "input → process → output" sequence in non-computer contexts through use of everyday examples. Suitable examples would be:  
   (i) Office clerk (in-tray → process → out-tray)  
   (ii) Producing class order of merit from examination results (collection of marks → processing marks according to given list of instructions → order of merit)  
   (iii) Film processing (film collected → processed → photographs)  
   The common feature of these examples should be identified: Stress the fact that the processing it carried out according to a definite set of instructions — the program. | A5. Any of the examples as a simulation exercise. |
| **2.2 The computer** The section is intended to give the student a general appreciation of how a computer system operates. Details of hardware are not required at this stage. The features to be stressed are:  
   (i) Input  
   (ii) Process — main store, arithmetic unit, control unit  
   (iii) Output | A6. Construction of a matchbox computer |
| **2.3 Simple programming** Students should work through simple programs where both program and data are stored in the matchbox computer. The following symbols should be used:  
   (i) Assignment instruction A ← B  
   (ii) A ← A + B  
   (iii) A ← A - B  
   (iv) A ← A * B  
   (v) A ← A / B  
   (vi) A ← A % B  
   (plus those of the form A ← A + B  
   A ← A - B  
   A ← A * B  
   A ← A / B  
   A ← A % B  )  
   (with parentheses) | A7. Use of the matchbox computer to illustrate simple computer and programming concepts. |
| **2.4 Hardware of a typical microcomputer system** The keyboard, VDU as an input device. The printer as an output device. The computer main store and the term random access memory (RAM) can be mentioned briefly, but details should not be given at this stage. Use of the microcomputer to run simple programs based on the instructions described in Section 2.3 should be demonstrated. | A8. Introductory demonstrations with the microcomputer |

---
## 3. FLOWCHARTING

### Syllabus

<table>
<thead>
<tr>
<th>Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
</table>
| 3.1 Flowcharting of non-computer problems | Analysis of everyday processes into a list of simple, sequential steps (e.g. starting a car, making tea etc.) Representation in flowchart form using the following symbols:  
(i) terminal box  
(ii) process box  
(iii) decision box  
(iv) flow lines Teachers should make sure that the examples chosen include decisions and loops. |
| 3.2 Computer algorithms in flowchart form | Introduction of the following symbols:  
(i) input/output box  
(ii) connector box Examples of computer flowchart algorithms should include area of circle, class test average, generation of Fibonacci sequence, binary search, bubble sort etc. Teachers should make sure that the examples chosen illustrate the use of unconditional branching, conditional branching and the use of a counter. The flowcharts used as examples should as far as possible be well-structured i.e. using only sequence, selection and iteration as building blocks. |
| 3.3 Conversion of simple flowcharts into BASIC | Through the use of simple flowchart examples, the following BASIC statements and operations should be introduced:  
(i) INPUT |
| 3.4 Importance of flowcharting | Teachers should ensure pupils appreciate the advantages of flowcharts i.e.  
(i) They clarify problems by breaking complicated processes into a series of simpler logical steps.  
(ii) They are easier to read than computer programs and are therefore a vital part of program documentation (See Section 4.1)  
(iii) They are an aid to debugging errors in algorithmic logic (by use of the "dry run"). |

---

A3. Students devise, debug and run their own simple programs on the microcomputer.
4. PROGRAMMING IN BASIC

<table>
<thead>
<tr>
<th>Syllabus</th>
<th>Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1 Program documentation</td>
<td>Students should be aware of the need for full program documentation and should maintain a well-documented file of their own programs. A fully-documented program should contain:</td>
<td></td>
</tr>
<tr>
<td>(i) title</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(ii) specification of problem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(iii) algorithm/flowchart</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(iv) list and definition of variables used</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(v) annotated program listing (using REM statements)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(vi) test data and sample output</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Students should realise from the commencement of their programming that the documentation must make the program understandable to the least knowledgeable potential user. It is expected that each student will devise and document during the course at least four separate programs of varying length, difficulty and subject matter.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.2 Program language</td>
<td>The version of BASIC studied should include the following:</td>
<td></td>
</tr>
<tr>
<td>(i) arithmetic operators (+, -, *, /)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(ii) relational operators (&lt;, &gt;, &lt;=, &gt;=, &lt;&gt;),</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(iii) logical operators (AND, OR, NOT)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(iv) arithmetic functions (SIN, COS, TAN, ATN, LOG, EXP, SQR, INT, RND, A$)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(v) string functions (LEN, LEFTS, MID$, RIGHTS, STRI$, VAL)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(vi) subprogram statements (PRINT, PRINT TAB, INPUT, DATA, READ, RESTORE)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(vii) program statements (DIM, STOP, END, GOTO, GOSUB, RETURN, FOR ..., TO ..., STEP, NEXT, IF ..., THEN)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Syllabus</th>
<th>Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>(vii) integer, real and string constants</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(viii) integer, real and string variables (simple and singly subscripted)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(ix) arrays (both one and two dimensions)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>In addition, students should know how to save, load, list, run and obtain hard-copy of programs. When writing their own programs students should be encouraged to use the full range of statements provided by the BASIC to which they have access.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.3 Programming</td>
<td>Students’ programs should not be solely concerned with mathematical problems. Students are expected to produce one fully documented program in at least two DIFFERENT areas. These might include</td>
<td></td>
</tr>
<tr>
<td>(i) games</td>
<td>A10. Individual computer program (see Notes).</td>
<td></td>
</tr>
<tr>
<td>(ii) data processing and business applications</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(iii) simulations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(iv) statistics and probability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(v) computer assisted learning instruction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(vi) graphing and plotting functions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(vii) mathematics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(viii) science</td>
<td></td>
<td></td>
</tr>
<tr>
<td>and so on.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. INPUT/OUTPUT, CODING AND THE STORAGE OF INFORMATION

<table>
<thead>
<tr>
<th>Syllabus</th>
<th>Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Coding in a non-computer context</td>
<td>The difference between information and its representation should be stressed; i.e. information is only the interpretation put on the data by people. Examples of non-computer coding of information should be discussed, e.g. Morse code, Braille, revolver colour codes, telephone area codes, students class numbers etc. The reasons for/advantages of coding should be considered. Students should be aware of the distinction between digital and analog systems of coding (e.g. digital and conventional watches).</td>
<td>A11 Conversion from one to another</td>
</tr>
<tr>
<td>5.2 Number codes</td>
<td>The representation of positive integers in denary, binary, octal and hexadecimal form.</td>
<td></td>
</tr>
<tr>
<td>5.3 Representation of numerical information within the computer</td>
<td>(The purpose of this section is to emphasise that numbers can be coded in different ways and, although human brains find the decimal system convenient, this need not be the case for machines.) This section is intended to give students an insight into the way that the computer systems, arithmetic and logic operations in binary code, integer and fractional binary numbers, addition and subtraction of positive binary numbers, sign bit, two's complement representation of negative numbers, Subtraction by addition of two's complements, range and accuracy of fixed word length representations of binary numbers. Basic ideas of truncation, rounding errors, overflow and floating point representation.</td>
<td></td>
</tr>
<tr>
<td>5.4 Coding of information in the computer main store</td>
<td>Students should appreciate that information (data and instructions) is stored and processed in binary code. The use of semiconductor RAM and ROM for the storage of binary words should be described, e.g. through a simple block diagram of a</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Syllabus</th>
<th>Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.5 Coding computer input</td>
<td>The need for coding should be stressed. Students should be familiar with the following ways of coding input data. (i) Hollerith punched cards (ii) bar codes (iii) VDU/keyboard terminals (students should realise that many keyboards generate ASCII code) (iv) magnetic ink and optical character recognition Codes should not be memorised. Typical applications of the above should be described, and the relative advantages, disadvantages discussed.</td>
<td>A12 Examination of the internal structure of a semiconductor RAM chip under a microscope (or a photograph of the same). A13 Demonstration of ASCII by POKEing codes into VIDE0 RAM locations. A14 Examination and decoding of sample cards and bar codes.</td>
</tr>
<tr>
<td>5.6 Decoding computer output</td>
<td>Students should be made aware that the decoding process is simply the reverse of the encoding process. The following peripheral devices should be introduced and discussed (i) line printer for hard-copy (golf ball, daz matrix, barrel types can be discussed) (ii) VDU.</td>
<td></td>
</tr>
<tr>
<td>5.7 Backing store</td>
<td>The distinction between the main (core) store and the backing store. Magnetic tape and magnetic disc as storage devices. Discussion of main/backup store with respect to (i) physical size (ii) storage capacity (iii) cost (iv) access modes (sequential/random) (v) access time for the different modes (vi) permanence.</td>
<td>A15 Audio monitoring of input/output from the microcomputer tape cassette deck.</td>
</tr>
</tbody>
</table>
6. DATA PROCESSING

6.1 Data processing concepts

This topic will deal with data processing without specific reference to computers at this time. The following areas will be covered:

(i) Definition of data and information from a commercial data processing standpoint.
(ii) Stages of data processing, which will briefly cover:
    - origin of data
    - data preparation
    - input of data to the system
    - processing of data
    - output of information

(iii) The concept of records, files and file storage from a commercial data processing standpoint. The record is described as a collection of related items and the file as a collection of related records.

6.2 Control of data

(i) This section will emphasise the importance of data being accurate and complete, and the consequences of error. This will be related to the previous section and will consider the problems of inaccuracy in:
    - Data capture
    - Data preparation and input
    - Processing of data

These points are to be considered from a commercial data processing standpoint and should be very closely related to the manual practical project on which the students are working.

(ii) Control of data in a computerized system should be discussed. Topics to be included should be:
    - Verification of data at the data preparation stage.
    - Validation of input by means of a program.
    - Error listings should be introduced here.
    - File security and copies of files and programs.

6.3 Electronic data processing

This section builds on the concepts developed earlier. The following topics should be discussed:

(i) Input devices. This should be a description of a device with which the students are familiar, together with a brief introduction to one other device for comparison. The emphasis here should be on the use of the device for data entry.

(ii) Output devices. Similar to (i)

(iii) File storage. Similar to (i)

(iv) File organization. A simplified description of sequential organization should be given, together with a brief overview of a direct access organization which can be demonstrated on the computer to which the students have access. A clear distinction should be drawn between file organization and access methods.

(v) Sorting of sequential files. The importance of sorting sequential files should be taught.

Practical Activities

A16b. The problems of data control should be brought into the practical project by the deliberate introduction of errors and omissions, so that the students can more readily appreciate the consequences of these problems.

A17. Parallel with this section the students should undertake a very simple computerized data processing project comprising a single master file and only one input record type with a maximum of three output reports, depending on the size of the project team.

If done in teams, the project will also illustrate the need for co-ordination and clear communication. The need for computer facilities should be minimized by this approach.

The project should deal with a simple school records system, to reduce initialization time. The project should include a data validation exercise based on the concepts discussed in 6.2.
## 7. COMPUTERS IN THE MODERN WORLD

<table>
<thead>
<tr>
<th>Syllabus</th>
<th>Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>In this section of the syllabus, a number of computer applications will be considered. They have been chosen because they serve to illustrate various modes of operation while not demanding too much background experience from the student. Teachers should ensure that the examples are discussed in terms of: (i) mode of operation (block diagram is sufficient) and flow of information; (ii) personnel involved; (iii) benefits and disadvantages (technological, economic and social).</td>
<td>A18 Teachers are encouraged to arrange class visits relevant to the applications in this section of the syllabus. Where possible, sample data, problems etc. should be obtained and discussed.</td>
</tr>
<tr>
<td>7.1 Batch processing</td>
<td>Students should be introduced to the savings involved if the mode of operation is changed, and the expense of information being out of date. Examples: (i) payroll; (ii) processing of customer orders.</td>
<td></td>
</tr>
<tr>
<td>7.2 Real-time processing</td>
<td>Students should be introduced to the support required for this mode of operation, e.g. the random organization and access of files, communication links. Examples: (i) banking; (ii) supermarkets.</td>
<td></td>
</tr>
<tr>
<td>7.3 Information retrieval</td>
<td>A particularly good real-time system that emphasizes on data structure and enquiry capabilities. Examples: (i) telephone enquiries; (ii) viewdata.</td>
<td></td>
</tr>
</tbody>
</table>

### Syllabus

#### 7.4 Dedicated computers

Many computer activities are performed by processors which are specifically designed (dedicated) for a single task. The advent of the micro-processor has made the computer more universally available for such single-purpose roles. Examples: (i) video games; (ii) automatic control of industrial processes.

#### 7.5 Computers and people

The implications of the rapid spread of computers on people and society. Examples: (i) The cashless society (autobank, credit/debit cards); (ii) The office and factory of tomorrow (word processors, robotics); (iii) The hospital of tomorrow; (iv) Patterns of employment and new lifestyles (extra leisure time, home computers); (v) Computer crime; (vi) Privacy.
8. COMPUTER OPERATION

<table>
<thead>
<tr>
<th>Syllabus</th>
<th>Notes</th>
<th>Practical Activities</th>
</tr>
</thead>
</table>

8.1 Control processing unit (CPU)

Students should know that the CPU contains the following interconnected units:

1. a program counter
2. an instruction register
3. control unit
4. an arithmetical logic unit (ALU)
5. accumulator(s)
6. other registers

The CPU communicates with the rest of the computer via an address bus, a data bus and a control bus.

8.2 Machine Code

Basic format of a machine instruction:
- Operation code
- Operand(s)

Composing a machine language program segment.

A19. Demonstration of simple machine code segment e.g. adding the content of a memory location to the content of an accumulator and placing the result in another memory location.

8.3 Program execution

Execution of a segment of a machine language program:

1. instruction fetch cycle
2. execution cycle

Explanation of the execution of the program segment in 8.2. Functions of the basic elements of the CPU can be re-emphasized here. The interconnections among these elements should be illustrated during the explanation.

8.4 Languages

Students should realize that the CPU always handles machine code. The relationship between machine code, assembly language, and high level languages, together with an understanding of the role of translators should be understood.
Marks System (Version 6.0)
Sung Tsan Middle School, Sai Kung
由真善其中学 微型電腦處理積分計算系統

簡介：本校在1982年9月開始設置電腦課程後，隨即計劃將電腦科技應用於學校行政上。其中包括設計及學生的成績。

我們曾用過現成的教務軟體如 VisiCalc 等，但因學生成績的計算方法繁瑣，輸入及輸出的格式與一般會計工作程序不同。發覺現有的一般軟體尚難滿足需求。

在這種情況下，我們着手設計一套適合我們教育工作者處理學生積分的軟件。經過無數次的修改及試用後，終於在83年9月正式使用在學校行政工作上，成效顯著。

83年初，本校於新學期開學時介紹使用本校軟件。由各老師及學生使用後，得以修正，而改今天的版本 (Version 6.0)。

系統要求：Apple II+ 64 K
基本上，一部磁碟機即可操作。

操作範圍：

功能包括：

1. 學生成績
2. 多次測驗成績
3. 考試平均
4. 每學年總成績
5. 學年總成績
6. 學年總成績
7. 考試平均
8. 多次測驗成績
9. 學生成績

積分的調整、修正、通知等，比較、統計，標準分的轉換，抽樣
不可日常稽查進行。
PART 1: GENERAL
A: EDIT FILES (open/retrieve/correct/revise)
B: FILE CATALOG (check/delete)
C: CALCULATION
D: SORTING
E: REVISE FILES
F: INSTALLATION
> OTHER PART (FOR RAMDISK USE ONLY)

PART 2: TERMS PRINT/GENERATE
PRINT
A: CLASS TERM RESULTS
B: AKNUAL TERM RESULTS
C: TEST/EXAM RESULTS
D: SUBJECT GRADING
E: TABULATION

GENERATE
F: ANNUAL TERM RESULTS
G: TEST/EXAM RESULTS
H: Z-SCORES
I: SUBJECT GRADINGS
J: INSTALLATION
> OTHER PART (FOR RAMDISK USE ONLY)

PART 3: FORM SEPARATION
A: SEPARATION by DTA FILES
B: EDIT SEPARATION FILE (open/retrieve/correct)
C: FORM SEPARATION
D: RETRIEVE CLASS NAMELIST
E: FORMAT FILES
F: REVOKE FILES
G: NAMELISTING BY NAMELIST
H: PRIZE/SUFFIX-NAMELIST
I: CATALOG CHECK
J: INSTALLATION

PART 4: BASIC DATA
A: WRITE NAME/STRU FILES
B: SUBJECT CODES (DEFINE/CORRECT)
C: PRINT TEACHER/SUBJECT CODES
D: PRINT NAMELIST (CLASS/FORM/SCHOOL)
E: PRINT STRUCTURE FILE
F: SORT CLASS NAMELIST TO FORM NAMELIST
G: GENERATE FORM NAMELIST TO FORMS-LIST
H: CATALOG CHECK
I: INSTALLATION

PART 5: COPY DISK
PART 6: DEMO DATA DISK
PART 7: REPORT
A: FORMATTING
B: GENERATOR
C: PRINT
D: CATALOG
E: INSTALLATION

PART 8: RAMDISK DRIVER
MKS SYSTEM VERSION 6.0
SUNG TSUN MIDDLE SCHOOL, SAI KUNG, N.T., HONG KONG

SPECIFICATIONS

No of Students/class : 50 max (25 char/length max)
(2 classes/sex sorting)
No of Teachers : 70 max
No of Subjects : 160 max (2 system defined codes included: TT: term:total, TV: term:average)
Reserved subject-codes:
also EN: English Language (EC/EO/ED/EG)
CI: Chinese Language (CC/CG)
and FL: structure file :FL3A.IER

STRUCTURE FILE : 30 files max

TERM RESULT : 25 FILES MAX
(TOTAL: value (99999) decimal place: 2
(AVERAGE: value (999.99) decimal place: 2
(SUBJECT: value (999) decimal place: 2

CALCULATION : 14 FILES MAX
(TOTAL: value (99999) decimal place: 0-2
(AVERAGE: value (999.9) decimal place: 0-2
(SUBJECT: value (999.9) decimal place: 0-1
character displayed on screen : 5

DATA FILE
SUGGESTION : 1 FORM (6 classes) 1 disk
-it depends on the sizes of the files.
FULL MARKS: 100
-scores may be scaled to 100 while editing.
50 VALUE/GRADE max/file
for numeric data: 0<N<99999 where N is the input value
decimal place: 0-2 places (suggest default 0)
for grade type data: 1-2 char .left justify choice
for those who are absent/non-elective, i.e. marked "--"
(average rate will not be counted for those "--")

HARDWARE EQUIPMENT

1) APPLE 2+ (with 16k ramcard)
APPLE 2x
2) 1-3 DRIVES
(RUN FAST by working with 128k/512k ramcard)
3) PRINTER (optional)
(RUN FAST by printing with buffer installed)
4) 80 col card will be used in PART 7 - REPORT
for checking / selecting PRINT-OUTS.

FOR FURTHER INFORMATION AND OTHER ENQUIRES
PLEASE CONTACT K.C. LEUNG, SUNG TSUN MIDDLE SCHOOL, SAI KUNG.
## Mathematics - Term 1 Test 1

### Lung Tsun Middle School, Sai Kung, H.K. (1985-86)

**File: MA3A.1T1**

**Date:** 04/04/86

**Class Teacher:** Leung Kam-chung

**Class Teacher:** Chu Hon-kwong

**Class Remarks:** Test Name List

**File Remarks:** Sample Test Data

**Assign Marks:** 40

**File Revised:** 190% - 100% (526315789)

**New File**

<table>
<thead>
<tr>
<th>No</th>
<th>Name</th>
<th>Scores</th>
<th>(Pos)</th>
<th>Raw Scores</th>
<th>REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cheng Chi-keung</td>
<td>100</td>
<td>(1)</td>
<td>100</td>
<td>190</td>
</tr>
<tr>
<td>2</td>
<td>Cheng Pik-fong</td>
<td>95</td>
<td>(4)</td>
<td>94.7368422</td>
<td>180</td>
</tr>
<tr>
<td>3</td>
<td>Cheung See-man</td>
<td>91</td>
<td>(6)</td>
<td>90.5263159</td>
<td>172</td>
</tr>
<tr>
<td>4</td>
<td>Cheung Yuk-lin</td>
<td>21</td>
<td>(18)</td>
<td>20.5263150</td>
<td>39</td>
</tr>
<tr>
<td>5</td>
<td>Chung Tai-shan</td>
<td></td>
<td></td>
<td></td>
<td>ABSENT</td>
</tr>
<tr>
<td>6</td>
<td>Chung Wei-hung</td>
<td>47</td>
<td>(17)</td>
<td>46.8421053</td>
<td>89</td>
</tr>
<tr>
<td>7</td>
<td>Ng Kam-kwong</td>
<td>97</td>
<td>(2)</td>
<td>96.8421053</td>
<td>184</td>
</tr>
<tr>
<td>8</td>
<td>Kong Yuk-chun</td>
<td>88</td>
<td>(7)</td>
<td>87.8947368</td>
<td>157</td>
</tr>
<tr>
<td>9</td>
<td>Kwok Chi-ming</td>
<td>84</td>
<td>(9)</td>
<td>83.9473684</td>
<td>159.5</td>
</tr>
<tr>
<td>10</td>
<td>Kwok Po-chun</td>
<td>71</td>
<td>(15)</td>
<td>70.5263158</td>
<td>134</td>
</tr>
<tr>
<td>11</td>
<td>Lai Fu-wing</td>
<td>0</td>
<td>(21)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>Lai Kwan-yip</td>
<td>76</td>
<td>(12)</td>
<td>76.3157895</td>
<td>145</td>
</tr>
<tr>
<td>13</td>
<td>Lau Yue-yip</td>
<td></td>
<td></td>
<td></td>
<td>ABSENT</td>
</tr>
<tr>
<td>14</td>
<td>Lee Chi-hung</td>
<td>88</td>
<td>(7)</td>
<td>87.8947368</td>
<td>167</td>
</tr>
<tr>
<td>15</td>
<td>Li Wei-kong</td>
<td>96</td>
<td>(3)</td>
<td>96.3157895</td>
<td>183</td>
</tr>
<tr>
<td>16</td>
<td>Ma So-ying</td>
<td>84</td>
<td>(9)</td>
<td>83.6842105</td>
<td>159</td>
</tr>
<tr>
<td>17</td>
<td>Ma Wai-lung</td>
<td>73</td>
<td>(14)</td>
<td>73.1576947</td>
<td>139</td>
</tr>
<tr>
<td>18</td>
<td>Man Wai-kong</td>
<td>68</td>
<td>(16)</td>
<td>67.8947368</td>
<td>129</td>
</tr>
<tr>
<td>19</td>
<td>Mui Wai-hung</td>
<td>76</td>
<td>(12)</td>
<td>76.3157895</td>
<td>145</td>
</tr>
<tr>
<td>20</td>
<td>Eek Siu-kwong</td>
<td>95</td>
<td>(4)</td>
<td>94.7368422</td>
<td>180</td>
</tr>
<tr>
<td>21</td>
<td>Yeung Kin-kai</td>
<td>82</td>
<td>(11)</td>
<td>82.1052632</td>
<td>156</td>
</tr>
</tbody>
</table>

- No OF Passes: 17 (89%)
- No OF FAILED: 2 (11%)
- No OF ABSENT: 2 (9%)

**Overall Average:** 75.37 (19)

**Verified:**

---

**Remarks:**
- Total number of candidates: 21
- Total number of passed: 17 (89%)
- Total number of failed: 2 (11%)
- Total number of absent: 2 (9%)

---

**Score Distribution:**
- 100% (1)
- 90% - 99% (4)
- 80% - 89% (6)
- 70% - 79% (9)
- 60% - 69% (2)
- 50% - 59% (1)
- 40% - 49% (1)
- 30% - 39% (1)
- Below 30% (1)

**Score Analysis:**
- Highest score: 100%
- Lowest score: 40%
<table>
<thead>
<tr>
<th>Test</th>
<th>Data 1</th>
<th>Data 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test A</td>
<td>12.3</td>
<td>45.6</td>
</tr>
<tr>
<td>Test B</td>
<td>78.9</td>
<td>23.4</td>
</tr>
<tr>
<td>Test C</td>
<td>56.7</td>
<td>89.0</td>
</tr>
</tbody>
</table>

*Note: Data is approximate and for demonstration purposes.*
(3A) 1ST TERM RESULT

SUNS TSUN MIDDLE SCHOOL, SAI KUNG, H.K. (1997-98)

SUN TSUN MIDDLE SCHOOL, SAI KUNG, H.K. (1997-98)

CLASS TEACHER: CHI Ho-tyung
CLASS REMARKS TEST RANKLIST

PASSING MARK: 50

<table>
<thead>
<tr>
<th>Roll No.</th>
<th>Name</th>
<th>Subjects</th>
<th>Total Ave.</th>
<th>CR</th>
<th>ER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CHENG Chi-kwun</td>
<td></td>
<td>299.41</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>2</td>
<td>CHEUNG Pi-tong</td>
<td></td>
<td>284.53</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>CHEUNG See-man</td>
<td></td>
<td>431.43</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>4</td>
<td>CHEUNG Si-tim</td>
<td></td>
<td>547.74</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>5</td>
<td>CHEUNG Si-sham</td>
<td></td>
<td>652.62</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>6</td>
<td>CHEUNG Si-lung</td>
<td></td>
<td>515.31</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>7</td>
<td>NG See-lee</td>
<td></td>
<td>452.60</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>8</td>
<td>KONG Si-chun</td>
<td></td>
<td>554.76</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>9</td>
<td>KONG Si-tim</td>
<td></td>
<td>500.77</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>10</td>
<td>KONG Si-sham</td>
<td></td>
<td>507.79</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>11</td>
<td>LEE Si-tim</td>
<td></td>
<td>399.55</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>12</td>
<td>LEE Si-chun</td>
<td></td>
<td>526.72</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>13</td>
<td>LEE Si-tong</td>
<td></td>
<td>524.74</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>14</td>
<td>LEE Si-ping</td>
<td></td>
<td>454.63</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>15</td>
<td>LEE Si-lung</td>
<td></td>
<td>448.62</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>16</td>
<td>NG Si-ping</td>
<td></td>
<td>370.51</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>17</td>
<td>NG Si-sham</td>
<td></td>
<td>504.59</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>18</td>
<td>NG Si-tim</td>
<td></td>
<td>412.61</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>19</td>
<td>NG Si-lung</td>
<td></td>
<td>452.62</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>20</td>
<td>NG Si-tong</td>
<td></td>
<td>432.14</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>21</td>
<td>NG Si-ping</td>
<td></td>
<td>428.56</td>
<td>22</td>
<td>22</td>
</tr>
</tbody>
</table>

Failed: 21 (103)
Passed: 131 (192)
Absent: 0 (00)

OVERALL AVERAGE: 62.74

FILE REMARKS:

SUB TEACHER: TSANG Chi-wai
1. CI3A.1ER (30/03/86): TEST-40% EXAM-60%
SUB TEACHER: TONG Kin-kwok
2. EN3A.1ER (21/21/21): AVERAGE FILE
SUB TEACHER: LEUNG Kam-chung
3. MA3A.1ER (04/04/86): TEST-40% EXAM-60%
4. SS3A.1ER (30/03/86): TEST-40% EXAM-60%
SUB TEACHER: WONG Kam-chung
5. PH3A.1ER (09/04/86): TEST-40% EXAM-60%
SUB TEACHER: TSANG Chi-wai
6. CI3A.1ER (30/02/86): TEST-40% EXAM-60%
SUB TEACHER: KWOK Yin-king
7. CM3A.1ER (20/03/86): TERM 1 EXAM
<table>
<thead>
<tr>
<th>FORM-CLASS</th>
<th>NAME</th>
<th>TOTAL</th>
<th>3A</th>
<th>3B</th>
<th>3C</th>
<th>3D</th>
<th>3E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-1</td>
<td>NG Yuen-ho</td>
<td>6</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2-1</td>
<td>WUNG Chun-fai</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3-1</td>
<td>MAN Chi-fai</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3-2</td>
<td>HUI Mei-sha</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3-2</td>
<td>YAU Kwok-wai</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4-1</td>
<td>CHEUNG Tze-ming</td>
<td>6</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4-2</td>
<td>LAU Siu-wan</td>
<td>7</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4-4</td>
<td>HO Kam-mui</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5-2</td>
<td>LI You-tok</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5-2</td>
<td>MA Choi-fung</td>
<td>10</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6-1</td>
<td>LIU Sik-kwong</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6-5</td>
<td>PHOEN Lai-har</td>
<td>12</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6-7</td>
<td>WU Sui-wan</td>
<td>13</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7-1</td>
<td>KWOK Chi-mino</td>
<td>14</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8-1</td>
<td>SOO Pui-man</td>
<td>15</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15-1</td>
<td>TSANG Chung-hong</td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17-2</td>
<td>CHIU Wai-brau</td>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17-2</td>
<td>FUNG Chui-fong</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19-4</td>
<td>CHUNG Wai-wing</td>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19-4</td>
<td>LING Mei-ye</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19-4</td>
<td>MAN Shu-vee</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-2</td>
<td>CHEUNG See-man</td>
<td>22</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-2</td>
<td>LAU Yuei-vinn</td>
<td>23</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-4</td>
<td>CHEUNG Siu-hung</td>
<td>24</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-4</td>
<td>LAI Sut-man</td>
<td>25</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-4</td>
<td>CHEUNG Siu-hong</td>
<td>26</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-4</td>
<td>TAM Ching-cheung</td>
<td>27</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-9</td>
<td>HO Cho-lung</td>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-9</td>
<td>SHEK Siu-wan</td>
<td>30</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22-9</td>
<td>YUENG Yin-oino</td>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32-4</td>
<td>MA Wai-ling</td>
<td>32</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32-12</td>
<td>WONG Kwok-tai</td>
<td>33</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34-5</td>
<td>CHEUNG Yuk-lin</td>
<td>34</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34-7</td>
<td>CHEUNG Yau-kik</td>
<td>35</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34-6</td>
<td>KONG Yuk-chun</td>
<td>36</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36-13</td>
<td>SHEK Siu-wan</td>
<td>37</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38-7</td>
<td>LEE Mei-ling</td>
<td>38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39-1</td>
<td>CHEUNG Mei-han</td>
<td>39</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39-1</td>
<td>MA Yuk-lin</td>
<td>40</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39-1</td>
<td>SUM kit-vee</td>
<td>41</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42-4</td>
<td>LAU Yim-fan</td>
<td>42</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42-4</td>
<td>LEE Shui-kuen</td>
<td>43</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42-4</td>
<td>WONG Tze-hua</td>
<td>44</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45-7</td>
<td>LEE kwok-bun</td>
<td>45</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45-7</td>
<td>LUNG Kwok-lin</td>
<td>46</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td>NAME</td>
<td>1ST RESULT</td>
<td>POS</td>
<td>40%</td>
<td>60%</td>
<td>TERM RAW RESULT</td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>-------------------</td>
<td>------------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----------------</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>CHENG Chi-keung</td>
<td>42 (19)</td>
<td>55.60</td>
<td>33.7</td>
<td>42.46</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>CHENG Pik-fong</td>
<td>56 (14)</td>
<td>57.86</td>
<td>55.4</td>
<td>56.384</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>CHEUNG See-man</td>
<td>78 (2)</td>
<td>75.52</td>
<td>80.2</td>
<td>78.328</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>CHEUNG Yuk-lin</td>
<td>76 (5)</td>
<td>55.51</td>
<td>89.1</td>
<td>75.664</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>CHUNG Lai-shan</td>
<td>31 (20)</td>
<td>31.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>CHUNG Wai-hung</td>
<td>59 (11)</td>
<td>66.87</td>
<td>53.5</td>
<td>58.848</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>HO Kam-kee</td>
<td>63 (9)</td>
<td>73.59</td>
<td>55.4</td>
<td>62.676</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>KONG Yuk-chun</td>
<td>74 (6)</td>
<td>69.20</td>
<td>77.2</td>
<td>74</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>KWOK Chi-ming</td>
<td>86 (1)</td>
<td>80.46</td>
<td>89.1</td>
<td>85.644</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>KWOK Po-chun</td>
<td>66 (7)</td>
<td>40.65</td>
<td>83.2</td>
<td>66.18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>LAI Fu-wing</td>
<td>60 (10)</td>
<td>67.21</td>
<td>55.4</td>
<td>60.124</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>LAI Kwan-yung</td>
<td>66 (7)</td>
<td>60.77</td>
<td>70.3</td>
<td>64.488</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>LAU Yue-yung</td>
<td>78 (2)</td>
<td>58.34</td>
<td>91.1</td>
<td>77.996</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>LEE Chi-hung</td>
<td>48 (17)</td>
<td>62.41</td>
<td>38.6</td>
<td>48.124</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>LI Wai-kong</td>
<td>58 (12)</td>
<td>77.53</td>
<td>44.5</td>
<td>57.712</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>MA So-ying</td>
<td>25 (21)</td>
<td>63.18</td>
<td>0.0</td>
<td>25.272</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>MA Wai-lung</td>
<td>77 (4)</td>
<td>59.22</td>
<td>88.1</td>
<td>76.548</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>MAN Wai-kong</td>
<td>47 (18)</td>
<td>67.35</td>
<td>33.7</td>
<td>47.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>MUI Wai-hung</td>
<td>58 (12)</td>
<td>60.78</td>
<td>55.4</td>
<td>57.552</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>SEK Sui-kwong</td>
<td>52 (15)</td>
<td>51.67</td>
<td></td>
<td>51.67</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>YEUNG Kin-kei</td>
<td>52 (15)</td>
<td>63.47</td>
<td>44.5</td>
<td>52.088</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

average: 60 61.05 59.3
no of PASSES: 16 19 13
no of FAILURES: 5 2 6
no of ABSENT: 0 0 2
<table>
<thead>
<tr>
<th>NO.</th>
<th>NAME</th>
<th>TOTAL ANNUAL RESULT</th>
<th>2/5</th>
<th>3/5</th>
<th>100%</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CHEUNG Chi-keung</td>
<td>219</td>
<td>44</td>
<td>42</td>
<td>45</td>
</tr>
<tr>
<td>2</td>
<td>CHEUNG Pak-fong</td>
<td>267</td>
<td>61</td>
<td>56</td>
<td>65</td>
</tr>
<tr>
<td>3</td>
<td>LEEING Set-sam</td>
<td>357</td>
<td>71</td>
<td>71</td>
<td>87</td>
</tr>
<tr>
<td>4</td>
<td>CHUANG Yiu-lin</td>
<td>411</td>
<td>94</td>
<td>76</td>
<td>89</td>
</tr>
<tr>
<td>5</td>
<td>CHEUNG Lei-shan</td>
<td>42</td>
<td>31</td>
<td>31</td>
<td>31</td>
</tr>
<tr>
<td>6</td>
<td>CHEUNG Kai-hung</td>
<td>305</td>
<td>78</td>
<td>60</td>
<td>78</td>
</tr>
<tr>
<td>7</td>
<td>LEE Shing</td>
<td>229</td>
<td>98</td>
<td>36</td>
<td>91</td>
</tr>
<tr>
<td>8</td>
<td>KEUNG Yiu-chun</td>
<td>217</td>
<td>43</td>
<td>74</td>
<td>22</td>
</tr>
<tr>
<td>9</td>
<td>KWOK Chi-sing</td>
<td>297</td>
<td>61</td>
<td>86</td>
<td>86</td>
</tr>
<tr>
<td>10</td>
<td>KWOK Po-chun</td>
<td>330</td>
<td>66</td>
<td>66</td>
<td>66</td>
</tr>
<tr>
<td>11</td>
<td>LEE Kwai-wing</td>
<td>354</td>
<td>71</td>
<td>60</td>
<td>78</td>
</tr>
<tr>
<td>12</td>
<td>LEE Chi-ling</td>
<td>132</td>
<td>66</td>
<td>66</td>
<td>66</td>
</tr>
<tr>
<td>13</td>
<td>LEE Ng-wai</td>
<td>415</td>
<td>65</td>
<td>78</td>
<td>99</td>
</tr>
<tr>
<td>14</td>
<td>LEE Chi-yung</td>
<td>257</td>
<td>71</td>
<td>42</td>
<td>57</td>
</tr>
<tr>
<td>15</td>
<td>LEE Wei-kong</td>
<td>317</td>
<td>62</td>
<td>56</td>
<td>67</td>
</tr>
<tr>
<td>16</td>
<td>MA Se-ying</td>
<td>152</td>
<td>26</td>
<td>25</td>
<td>44</td>
</tr>
<tr>
<td>17</td>
<td>MA Ka-tsing</td>
<td>222</td>
<td>64</td>
<td>77</td>
<td>95</td>
</tr>
<tr>
<td>18</td>
<td>MAW Hai-kong</td>
<td>295</td>
<td>59</td>
<td>47</td>
<td>67</td>
</tr>
<tr>
<td>19</td>
<td>MAW Hai-kong</td>
<td>377</td>
<td>75</td>
<td>53</td>
<td>87</td>
</tr>
<tr>
<td>20</td>
<td>SIA Sin-kwong</td>
<td>305</td>
<td>61</td>
<td>52</td>
<td>67</td>
</tr>
</tbody>
</table>

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>TEUNG Kin-kai</td>
<td>271</td>
<td>74</td>
<td>52</td>
<td>89</td>
</tr>
</tbody>
</table>

**Average:**
- 64
- 60
- 69

**No. of Passes:**
- 17
- 16
- 15

**No. of Failures:**
- 4
- 5
- 4

**No. of Absent:**
- 0
- 0
- 2
## Mathematics - Classes Grading

### Sung Sun Middle School, Sai Kung, H.K. (1997-98)

**File: MASC.1ER**
**Date: 04/04/86**

**S/D, Teacher:** HO Lai-mui  
**Class Teacher:** CHAN Shun-ya

**File Remarks:** (20/03/86) 2nd Term Exam  
**Class Remarks:** Name List File

<table>
<thead>
<tr>
<th>NO</th>
<th>Name</th>
<th>Grade</th>
<th>Marks</th>
<th>Raw Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CHAN F.;Jirm-an</td>
<td>27.54</td>
<td>27.54</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>CHAN Yu.-:u&quot;, r</td>
<td>45.36</td>
<td>45.36</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>CHEUN C••••••••••••••••</td>
<td>63.18</td>
<td>63.18</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>CHUNG Mei-han</td>
<td>72.90</td>
<td>72.9</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>CHIU Fai-yeung</td>
<td>--</td>
<td>(ABSENT)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>CHUNG Siu-lan</td>
<td>34.02</td>
<td>34.02</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>HO Chun-kwong</td>
<td>27.54</td>
<td>27.54</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>HO Yuk-mai</td>
<td>45.36</td>
<td>45.36</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>KAN Shuk-ling</td>
<td>61.93</td>
<td>61.93</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>LAU Siu-wong</td>
<td>43.74</td>
<td>43.74</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>LAU Yin-fan</td>
<td>72.09</td>
<td>72.09</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>LEE Kwok-bun</td>
<td>70.47</td>
<td>70.47</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>LEE Shui-kuen</td>
<td>72.09</td>
<td>72.09</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>LEUNG Sau-lin</td>
<td>70.47</td>
<td>70.47</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>LIU Yee-lan</td>
<td>52.65</td>
<td>52.65</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>LO Hoi-yin</td>
<td>27.54</td>
<td>27.54</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>LO Lei-chun</td>
<td>45.36</td>
<td>45.36</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>NA Siu-kuen</td>
<td>63.18</td>
<td>63.18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>NA Yuk-lin</td>
<td>72.09</td>
<td>72.09</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>NA Wai-chung</td>
<td>25.11</td>
<td>25.11</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>NG Man-yin</td>
<td>17.82</td>
<td>17.82</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>NG Kwok-tong</td>
<td>27.54</td>
<td>27.54</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>SHK Kui-chun</td>
<td>53.46</td>
<td>53.46</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>SI Siu-mei</td>
<td>63.18</td>
<td>63.18</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>SI Chi-yee</td>
<td>72.90</td>
<td>72.9</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>SI Ko-tsun</td>
<td>60.75</td>
<td>60.75</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>SI Yu-ku-mei</td>
<td>54.27</td>
<td>54.27</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>SI Yip Shiu-hua</td>
<td>72.09</td>
<td>72.09</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>SI Yip Kiu-tai</td>
<td>68.85</td>
<td>68.85</td>
<td></td>
</tr>
</tbody>
</table>

### Statistics

<table>
<thead>
<tr>
<th>Grade</th>
<th>No of Students (%)</th>
<th>Grade</th>
<th>Acc. No (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>0.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>D</td>
<td>14</td>
<td>50.00%</td>
<td>50.00%</td>
</tr>
<tr>
<td>E</td>
<td>17</td>
<td>60.71%</td>
<td>60.71%</td>
</tr>
<tr>
<td>F</td>
<td>20</td>
<td>71.43%</td>
<td>71.43%</td>
</tr>
<tr>
<td>G</td>
<td>22</td>
<td>78.57%</td>
<td>78.57%</td>
</tr>
<tr>
<td>H</td>
<td>28</td>
<td>100.00%</td>
<td>100.00%</td>
</tr>
</tbody>
</table>

Total: 49  
Lower limit is inclusive
## Class Teacher: Chiu Hon-kwong

## Class Remarks: Test Name List

### Overall: 62.73 (21/21)

### File Remarks:

1. **C3A.1EG (11/22/33)**: Grading - C3A.1ER
   - **SUB Teacher**: TSANG Chiu-wai

2. **EN3A.1EG (28/03/86)**: Grading - EN3A.1ER
   - **SUB Teacher**: TONG Kin-kwok

3. **MA3A.1EG (04/04/86)**: Grading - MA3A.1ER
   - **SUB Teacher**: LEUNG Kam-chung

4. **SS3A.1EG (30/03/86)**: Grading - SS3A.1ER
   - **SUB Teacher**: WONG Kam-cheung

5. **PH3A.1EG (30/03/86)**: Grading - PH3A.1ER
   - **SUB Teacher**: TSANG Chiu-wai

6. **C3A.1EG (11/22/33)**: Grading - C3A.1ER
   - **SUB Teacher**: CHU Hon-kwong

7. **CM3A.1EG (30/03/86)**: Grading - CM3A.1ER

8. **TV3A.1ER (04/04/86)**: Average File (Term Result)

9. **TT3A.1ER (04/04/86)**: Total File (Term Result)
OBJECTIVES:
1. To provide opportunity for the study of modern methods of information processing so that students may understand and apply this rapidly growing technology.
2. To acquaint students with the uses and limitations of computers.
3. To develop among students problem solving skills through interaction with computers.
4. To encourage an understanding of the implications of computers in the modern world.
5. To prepare students who wish to go on to further studies in computer science.

TIME ALLOCATION:
In accordance with the E.D. recommendation, 4 periods per week are allocated to the actual teaching of this subject at both F.4 and F.5. Since the students are taking this subject as an extra subject in addition to the normal eighth subjects, these lessons take place after normal schools hours. Besides during the lessons, the students have on average about one and a half hour of practice time at the computer room.
Since it is desirable that students can have more time to practice using each new command as they are being taught, Basic Programming should be started as early as possible. Moreover, the teaching is divided into two sections from October onwards - half of the lessons are devoted to Basic Programming and half to basic computer theory.

<table>
<thead>
<tr>
<th>Month</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sept.</td>
<td>Input -- process -- output concepts</td>
</tr>
<tr>
<td></td>
<td>The Computer</td>
</tr>
<tr>
<td></td>
<td>Simple Programming concepts</td>
</tr>
<tr>
<td></td>
<td>Flowcharting of non-computer problems</td>
</tr>
<tr>
<td></td>
<td>Hardware of a typical microcomputer system</td>
</tr>
<tr>
<td>Oct.</td>
<td>BASIC Programming</td>
</tr>
<tr>
<td></td>
<td>A. Sequential statements</td>
</tr>
<tr>
<td></td>
<td>Arithmetic operators</td>
</tr>
<tr>
<td></td>
<td>Variable/data</td>
</tr>
<tr>
<td>Nov.</td>
<td>B. Branch Statements</td>
</tr>
<tr>
<td></td>
<td>Relational/logical operators</td>
</tr>
<tr>
<td></td>
<td>C. Loop Statements</td>
</tr>
<tr>
<td>Dec.</td>
<td>D. Arrays</td>
</tr>
<tr>
<td></td>
<td>E. Subroutines</td>
</tr>
<tr>
<td>Jan.</td>
<td>F. Functions</td>
</tr>
<tr>
<td>Feb.</td>
<td>G. Project 1 (to be handed in after Easter)</td>
</tr>
<tr>
<td>Mar.</td>
<td>H. Sorting</td>
</tr>
<tr>
<td>Apr.</td>
<td>I. Project 2 (to be handed in before summer vacation)</td>
</tr>
<tr>
<td></td>
<td>J. Filing Statements</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Computer Theory</td>
</tr>
<tr>
<td></td>
<td>1.1 Calculating aids</td>
</tr>
<tr>
<td></td>
<td>1.2 Mechanical stored program devices</td>
</tr>
<tr>
<td></td>
<td>1.3 Mechanical data storage devices</td>
</tr>
<tr>
<td></td>
<td>1.4 The development of electronic computers</td>
</tr>
<tr>
<td></td>
<td>3.2 Computer algorithms in flow-chart form</td>
</tr>
<tr>
<td></td>
<td>3.3 Conversion of simple flowcharts into BASIC</td>
</tr>
<tr>
<td></td>
<td>3.4 Importance of flowcharting</td>
</tr>
<tr>
<td></td>
<td>3.2 Computer algorithms in flow-chart form</td>
</tr>
<tr>
<td></td>
<td>3.3 Conversion of simple flowcharts into BASIC</td>
</tr>
<tr>
<td></td>
<td>3.4 Importance of flowcharting</td>
</tr>
<tr>
<td></td>
<td>4.1 Program documentation</td>
</tr>
<tr>
<td></td>
<td>5.1 Coding in a non-computer context</td>
</tr>
<tr>
<td></td>
<td>5.2 Number Codes</td>
</tr>
<tr>
<td></td>
<td>5.3 Representation of numerical information in the computer</td>
</tr>
<tr>
<td></td>
<td>5.4 Coding of information in the computer main store</td>
</tr>
<tr>
<td></td>
<td>5.7 Backing store</td>
</tr>
</tbody>
</table>
May
J. Filing Statements

June
K. Project 3 (to be handed in after summer vacation)

Teaching Schedule for F.5

Sept.
Project 4 (to be handed in after X'mas vacation)
7.1 Batch processing
7.2 Real time processing

Oct.
7.3 Information Retrieval
7.4 Dedicated computers
7.5 Computers and people

Nov.
Project 4
8.1 Central Processing Unit (CPU)
8.2 Machine Code
8.3 Program execution

Dec.
8.4 Languages
Project 4

Jan.
Review and discussion of the projects submitted by students. Modification and upgrading of projects if applicable. Students should be able to learn much from studying each other's project.

Feb.
Data Processing
Input/output, coding and the storage of information

Mar.
Computers in the Modern World
Revision
INTRODUCTORY NOTE

The objective of this course is to introduce to the students the concept of the stored-program computer, followed by a detailed study of a high-level programming language, PASCAL.

At the end of this course, students will be able to use the high level language in statistical and numerical applications.

PRE-REQUISITE

Nil.

UNIT VALUE and DURATION

1 unit; 60 hours (45 hours lecture, 15 hours tutorial)

SYLLABUS

C1.1 Concept of a Stored-Program Computer (10 hours)

Basic parts of a computer; input/output, control, storage, arithmetic and logical units. Elementary concepts of auxiliary storage unit.

C1.2 High-level Language (40 hours)

Introduction to PASCAL. Use of flow charts, syntax diagrams. Elements of problem analysis; construction and testing of algorithms. Data vetting and techniques for program recovery following detection of input data errors.

Structured programming in PASCAL, procedures and functions. The scope of identifiers, global variables, scalar types and subrange types. Structured data types, arrays, records, files sets and compound structures.
Cl-3 Systems Software Concept (10 hours)

Introduction to operating system, multi-tasking/time sharing operating system, loader, assembler, interpreter, and compiler. Use of applications packages, library routines and published algorithms.

ASSESSMENT

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Examination</td>
<td>70%</td>
</tr>
<tr>
<td>Continuous assessment</td>
<td>30%</td>
</tr>
<tr>
<td>Total</td>
<td>100%</td>
</tr>
</tbody>
</table>

BIBLIOGRAPHY

Essential

Findlay, W, Watt, D.A.  
PASCAL. An Introduction to Methodical Programming, 3rd Ed.  
Pitman 1985

Supplementary

Graham, N.  
Introduction to PASCAL  
West 1984

Manual on PASCAL
INTRODUCTORY NOTE

The objective of this course is to give a functional understanding of how a computer works and interfaces to external devices. A bottom up approach is taken starting from the primitive devices used in the construction of the various subsystems which are then used to build the final computer system.

PRE-REQUISITE

23204 PROGRAMMING AND COMPUTER ORGANISATION I

UNIT VALUE and DURATION

1 unit; 60 hours (45 hours lecture, 15 hours tutorial)

SYLLABUS

C2-1 Representation and Storage of Data (12 hours)

The computer's memory, computer words, contents and interpretation. Representation and manipulation of information, boolean, numeric, ASCII Codes. Code conversion. Number bases and conversion.

Sign-and-modulus and two's complement notation, arithmetic operations. Logical operations and truth tables. Floating-point numbers and manipulation.

C2-2 Concepts of Computer Organisation (20 hours)

Primitive devices; gates, flip-flops, wires, buses, registers, multiplexers. Arithmetic and logical unit; accumulator, complement, increment, adder, test for zero. Memory; address selection, read invite, RAM/ROM. Control unit; instruction register, data register, fetch/execute sequence. Clock. Microprogramming.
C2-3 Peripheral Communications (14 hours)


C2-4 FORTRAN Programming (14 hours)

Recursion and its uses. Programming techniques for sorting, searching and merging.


ASSESSMENT

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Examination</td>
<td>70%</td>
</tr>
<tr>
<td>Continuous Assessment</td>
<td>30%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

BIBLIOGRAPHY

**Essential**


**Supplementary**

23206 MICROCOMPUTERS AND APPLICATIONS

INTRODUCTORY NOTE

The objective of this course is to familiarise the students with popular and updated microcomputer application software. The course is practically orientated with 50% hands-on supervision.

The course also develops student's ability to use word processing and graphic facilities in preparing project reports and other documents.

PRE-REQUISITE

23204 PROGRAMMING AND COMPUTER ORGANISATION I

UNIT VALUE and DURATION

1/2 unit; 30 hours (15 hours lecture, 15 hours practical)

SYLLABUS

C3-1 Introduction (6 hours)

Evolution of microcomputer technology and functional characteristics. Introduction to the microcomputer; familiarity with keyboard and screen editing; floppy disk media; menu driven interface; user friendliness; turnkey systems; taking a "back-up"; DOS commands; use of popular integrated applications softwares.

C3-2 Spreadsheet (6 hours)

Spreadsheet fundamentals for tabular manipulation; typical applications; survey and demonstrations.

C3-3 Word Processing (6 hours)

Word processing fundamentals for textual manipulation; practice using a word processing package; survey and demonstrations. Integrated systems; survey.
C3-4 Database Systems (6 hours)

The database approach for record manipulation; data management systems; survey and demonstrations.

C3-5 Graphics (6 hours)

Statistical graphics; use of graph plotter and graphics printer.

ASSESSMENT

Examination 50%
Continuous assessment 50%
Total 100%

BIBLIOGRAPHY

Essential

The Lotus Guide to Symphony Command Language Programming Techniques
Addison-Wesley 1985

Supplementary

Reference manual on Symphony
Lotus 1984
INTRODUCTORY NOTE

This course consists of two major aspects of information processing. Students will be exposed to both theory and practice on these aspects of commercial computing.

PRE-REQUISITES

23204 PROGRAMMING AND COMPUTER ORGANISATION I
23206 MICROCOMPUTERS AND APPLICATIONS

UNIT VALUE and DURATION

1 unit; 60 hours (45 hours lecture, 15 hours tutorial)

SYLLABUS

C4-1 Introduction to COBOL Programming (22 hours)

Program syntax and structure.

Validity checking: Record manipulation and application to statistical survey.

Table handling, sorting and merging.

Sequential file and index file.

Documentation and maintenance.

C4-2 Introduction to 4GL (4 hours)

Concept of 4GLs. Demonstration and use of Powerhouse package.

C4-3 Database Concepts (16 hours)

Data Base Management System (DBMS), role of data base administrator, data base models, including relational database model, evaluation and selection of data base systems.
C4-4 **Introduction to Systems Analysis and Design**  (10 hours)

Systems analysis concepts. The systems life cycle. Structured methodologies e.g. SSADM. System investigation techniques. Systems design concepts; input, output, file and procedure design. Justification and implementation.

C4-5 **Simple Project**  (8 hours)

Small project or case study relating COBOL, database and/or system analysis and design.

**ASSESSMENT**

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Examination</td>
<td>60%</td>
</tr>
<tr>
<td>Continuous assessment</td>
<td>40%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

**BIBLIOGRAPHY**

**Essential**

Popkin, G.S.  Comprehensive Structured COBOL, 2nd edition  Kent 1986

Date, C.J.  An Introduction to Database System, 4th edition  Addison-Wesley 1986

**Supplementary**

Yeates, D.  Basic Systems Analysis  Pitman 1984

Daniels, A., Yeates, D.  Practical Systems Design  Pitman 1984
23314 INTRODUCTION TO SYSTEMS SOFTWARE

INTRODUCTORY NOTE

The aim of this course is to provide the students with a comprehensive knowledge of the internal computer operation and the interface between hardware and software.

PRE-REQUISITES

23204, 5 PROGRAMMING AND COMPUTER ORGANISATION I, II

UNIT VALUE and DURATION

1 unit; 60 hours (45 hours lecture, 15 hours tutorial)

SYLLABUS

C5-1 Assembly Language Programming (30 hours)

Basic computer structure, addressing modes. Machine level instructions, input/output/file operations. Programming techniques; reentrancy and recursion. Linkage with high level languages.

C5-2 Operating Systems and Systems Software (30 hours)

CPU/storage/device management. Deadlock problems and solution.

Case study on UNIX System. Design goals, process control. I/O system. File system and shell.

Basic functions and machine-dependent/independent features of loaders, linkers, assemblers and compilers.
23314 (2)

ASSESSMENT

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Examination</td>
<td>70%</td>
</tr>
<tr>
<td>Continuous assessment</td>
<td>30%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

BIBLIOGRAPHY

**Essential**


**Supplementary**

Deitel, M. *An Introduction to Operating Systems* Addison-Wesley 1984

Beck, L. *System Software: an introduction to systems programming* Addison-Wesley 1985
INTRODUCTORY NOTE

The objective of this course is to provide the student with fundamentals of data communications and networking, a subject of increasing importance.

In teaching this subject, the background of the students should be noted in order to avoid unnecessary technicality in engineering knowledge.

PRE-REQUISITE

23204 PROGRAMMING AND COMPUTER ORGANISATION I

UNIT VALUE AND DURATION

1 unit; 60 hours (45 hours lecture, 15 hours tutorial)

SYLLABUS

C6-1 Data Communications (30 hours)

Data transmission, synchronisation, error detecting methods. Link-level protocols, error control, flow control, link management. The electrical interface, transmission media, signal types, physical level protocols. Terminal based networks, terminal characteristics, character-mode networks, block-mode networks, terminal network protocols.

C6-2 Computer Networks (30 hours)

ASSESSMENT

Examination 70%
Continuous Assessment 30%
Total 100%

BIBLIOGRAPHY

Essential

Halsall, F. Introduction to Data Communications & computer networks Addition Wesley 1985
INTRODUCTORY NOTE

This course is of practical nature. Students are required to complete programming projects on specified topics integrating materials from previous courses.

PRE-REQUISITES

23313 INTRODUCTION TO INFORMATION PROCESSING
23314 INTRODUCTION TO SYSTEMS SOFTWARE
23423 FUNDAMENTALS OF DATA COMMUNICATIONS

UNIT VALUE AND DURATION

1 unit; 60 hours (Seminar and Project)

SYLLABUS

C7-1 Seminar (20 hours)
Seminar or research activities on general concepts of software engineering.

C7-2 Programming Project (40 hours)
Writing of one large program or several shorter programs representing a total of approximately 40 hours of programming time.

In this programming project, the students are expected to demonstrate their knowledge of some of the following aspects of programming:

1. Software design;
2. Program verification and validation;
3. User interface consideration;
4. Software documentation;
5. Software maintenance.

ASSESSMENT

Continuous assessment 100%

BIBLIOGRAPHY

(As specified by Lecturers)
APPENDIX C
Class: _______________________

1. Sex: Male / Female

2. What is your age group?
   - [ ] Below 20
   - [ ] 20 - 21
   - [ ] 22 - 23
   - [ ] 24 or above

3. How much do you think you know about computer?
   - [ ] nothing
   - [ ] a little
   - [ ] a few
   - [ ] fair
   - [ ] well
   - [ ] very well

4. Have you ever used a computer or an item that is controlled by computer?
   - [ ] Yes
   - [ ] No

5. What advantage(s) do you think a computer has
   - [ ] fast in speed
   - [ ] accurate in calculation
   - [ ] can solve problems for you
   - [ ] easy to use
   - [ ] no need to take care of
5. An ordinary person has I.Q. around 100. What I.Q. score would you give to a computer?

7. What things you would like to be processed by computer?

8. Name 5 computer languages, spell them in full if you can.

9. Name 5 computer manufacturers in full name, give 1 computer model for each manufacturer if you can.
APPENDIX  D
Topic: Error

Keywords
Mantissa, truncation error, rounding error, absolute error, relative error, percentage error, propagation of errors.

Objectives
After working through this section, you should be able to:

(i) identify what error, in general, may occur when calculating machines are applied on numerical data

(ii) find the maximum propagation absolute error, relative error and percentage error when addition, subtraction, multiplication and division are operated on approximated values
Learning activity

1. Introduction

Before we talk about errors, it is better to understand the representation of numbers inside a computer or calculator. Numbers are usually represented as

```
| sign | exponent | mantissa |
```

The number could be stored in binary form, hexadecimal form, or packed decimal form. The way numbers are stored depends purely on the design of the computer or calculator, we are not going to discuss them in detail, interested readers may find these materials in most elementary numerical text books. However, no matter what sort of system is used, all stored numbers have to have finite number of digits. Some may have 8 digits, some may have 11 digits, some may have 16 or more. Anyway, as long as only finite number of digits is possible, error is expected to occur in representing some numerical values. For example, irrational numbers, such as $\sqrt{2}$, some fractions, such as $1/3$, cannot be expressed in finite number of digits.

There are many possible sources of errors, and errors obviously affect the accuracy of the solution of a particular problem. If we know that there is error in our solution, no matter how large it is, we may alert it and at most discard the solution. But the worst thing is that we do not know that has error and accept the garbage.

When manipulation of data is needed, error is possible to occur. In general, errors can be induced in three ways:

(i) Errors due to nature of data and data transfer.
(ii) Errors due to truncation and rounding.
(iii) Errors due to inappropriate interpretation of formulas.

We are going to discuss them in detail.
2. Errors Due to the Nature of Data and Data Transfer

Some obtained data may have error caused by uncertainty in measurements, accuracy of instruments, or factors affected by uncontrolled environment. Errors of these types cannot be corrected because data may come from experiments or observations. What could possibly be done is to analyse the obtained data using statistical techniques to identify the errors, or to repeat the experiment or observation if possible, hoping that errors can be averaged out.

In the case when data are needed to be transferred, copied or keyed in, many devices, such as mark sense card, magnetic or optics sensor, bar codes, have been developed to reduce the possibility of making errors. If data must be punched in by operators, other than double checking, two persons can be used to input the same data, and signals are given if the two inputs do not match. In numerical data, they can also be validated by applying range check, consistency check, or check digit to minimize the chance of getting error.

(3) Error Due to Truncation and Rounding

By the first principle of calculus, it is given that

\[
\frac{dy}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}
\]

In theory, the value of \( h \) has to approach to zero. If we apply this formula numerically, then the smaller value in \( h \) we use, the better approximation the derivative it should be. Let us look at the following example:-

Let \( f(x) = \frac{e^x}{x} \), implies \( f'(x) = \frac{x-1}{x^2} e^x \).

For \( x = 20 \), 
\[
f'(20) = 23045346.78
\]

Using the first principle of calculus

\[
f'(20) = \frac{e^{20+h} - e^{20}}{20+h - 20} \]

Let
\[
\begin{align*}
h & = .1 & f'(20) & = 24178827.40 \\
h & = .001 & f'(20) & = 23056330.00 \\
h & = .0001 & f'(20) & = 23046500.00 \\
h & = .00001 & f'(20) & = 23046000.00 \\
h & = .000001 & f'(20) & = 23050000.00 \\
h & = .0000001 & f'(20) & = 23000000.00
\end{align*}
\]
The above calculation performed on an 10-digit calculator (HP-41C). It can be seen that the result gets better at first, but as h is getting too small, the result gets worse and worse. This is due to the fact that when h is getting too small, $e^{20+h}$ and $e^{20}$ are very close to each other, their difference, which may not be very exact, is divided by a very small number h, this magnifies the error even more. (More details concerning error in subtraction shall be discussed later.) Of course, how serious the error would be depends on what kind of machine you use, but the function itself also plays an important role too.

Let us look at two more examples to demonstrate the effect of truncation and rounding errors that may occur by computer.

The first example is to sum $1/N$ N times for $N=100$. It is obvious that the result should be 1. However, using single precision, the computer gives 0.999999 (Appendix I). The correct result can only be obtained by using double precision (Appendix II).

The second example uses the computer to solve a system of linear simultaneous equations by Cramer's rule. The given system has two equations and two unknowns:

$$
0.2038X + 0.1218Y = 0.2014 \\
0.4071X + 0.2436Y = 0.4038
$$

It can easily be verified that the solution set is $X = -2$ and $Y = 5$. A program is written to solve this particular problem. It is as expected that the exact result cannot be obtained by using single precision (Appendix III). However it only improves a bit by using double precision (Appendix IV). But if we modify the system by multiplying the two equations by 10000 (we should still obtain the identical property and have the same solution set) and then solve it again, surprisingly, it gives the exact solution even though only single precision is used (Appendix V).

As the conclusion, rounding error and truncation error can affect the results no matter how simple the calculation is. Errors can be minimized if we can avoid subtracting two numbers that are very close in magnitude from each other, avoid adding two numbers that are too far in magnitude from each other, and use integral numbers whenever possible. Last, but most important, apply the theories to support your answers.
4. Error Due to Inappropriate Interpretation of Formulas

This type of error depends very much on the nature of the problem itself, hence it is not possible to bring out all the problems in question. However, some examples are given as guidelines.

In doing problems such as probability, it may come across the factorials, such as, the combination of n objects, taken r at a time, is given by \( nC_r = \frac{n!}{r!(n-r)!} \). If we evaluate all the factorials first, then perform the multiplication and division, as the way the formula stated, for large n, we may overflow a machine easily, because if \( n>69 \), \( n!>10^{99} \), which exceeds the limit of the magnitude that most calculators can hold.

An example is given in Appendix VI, which applies the formula directly and obtains an overflow message in trying to evaluate \( 34C3 \). However, if the formula is rewritten as

\[
\frac{n!}{r!(n-r)!} = \frac{n(n-1)(n-2)\ldots(n-r+1)}{r!}\]

the correct value 5984 can then be obtained (Appendix VII).

Other than the overflow problem, we may sometimes get a slight rounding error due to considerably large number of multiplications. For example, the correct answer of \( 38C19 \) is 35,345,263,800. However, if we apply the formula by using a calculator, we may get 3.5345263 x 10^{10} or at most 35,345,263,810. Therefore, in performing calculations of this nature, we should rearrange the terms so that some numbers can cancel one another in order to eliminate errors as far as possible.

We have mentioned that we should avoid adding a very small number to a very large number. However in evaluating functions defined by convergent infinite series, such as \( \sin x \) or \( \log x \), we always confront two problems. Firstly, we know that it is not possible to add infinitely many terms of an infinite series, we can only compute the nth partial sum, hence only the approximation of the infinite series can be found which bound to have truncation error. Secondly, by the necessary condition of a convergent infinite series, the limit of the nth term of the series, as n approaches to infinity, has to be zero.
That is to say, in evaluating an infinite series, we would, eventually, be adding a very very small number (approaching zero) to a comparatively large number. In this case, no matter how many more terms we have added, the result would not be improved. For example,

\[ S = \sum_{n=0}^{\infty} \frac{1}{4^n} \]

\[ = 1 + \frac{1}{4} + \frac{1}{16} + \frac{1}{64} + \frac{1}{256} + \ldots \]

when \( n = 10 \), the sum of first 10 terms becomes 1.333332062 and the 11th term, \( \frac{1}{4^{10}} = 0.0000009536743164 \), and most of the significant figures would not be added into it. One way to minimize this kind of error is by summing the infinite series backwards, that is, adding the small terms first. But the drawback of this method is that the number of terms to be added in order to produce the best result have to be tested and be determined before this method can be applied. However, if the characteristics of the infinite series can be found by some known theorems, such as the theorem by Leibnitz*, the number of terms to be used can be pre-determined.

For example, the infinite series

\[ S = 1 - \frac{1}{2} + \frac{1}{4} - \frac{1}{8} + \frac{1}{16} - \ldots \]

if we want to evaluate \( S \) correct to 3 decimal places, because \( \frac{1}{2^{10}} \) is less than .001, therefore we can compute \( S \) by summing 10 terms backwards.

Moreover, many functions or transcendental numbers can be interpreted by more than one infinite series. Of these infinite series, some converge very nicely, but some do not. If possible, choose a well behave infinite series to work with so as to reduce errors. For example,

\[ \frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \ldots \]

To calculate \( \pi/4 \) correct to 6 decimal places using this series, it would need more than 1,000,000 terms. With this large amount of computation, the round-off error in the calculation itself is probably much larger. Another series to compute \( \pi \) is therefore much desired.
Another example concerning the rearranging of formulas to reduce the possibility of computational error is given as follow. Sometimes in function evaluation we would be involved in calculating high power polynomials, such as

\[ P(x) = 12x^6 + 10x^5 + 8x^4 + 6x^3 + 4x^2 + 2x + 1 \]

For large values of \( x \), if we evaluate \( P(x) \) from left to right, probably the last few terms would not be added into it because they may be comparatively too small. A factorization method, called the Horner's rule, can rearrange a polynomial into a way that not only the error mentioned above can be minimized, it also reduces the number of calculations involved. The method is to take out the higher powers of \( x \) one by one and rewrite \( P(x) \) as

\[ P(x) = (((((12x + 10)x + 8)x + 6)x + 4)x + 2)x + 1. \]

Let us demonstrate the errors with one more simple example. Use an electronic calculator (10-digit calculator) to do the following two problems, summing accordingly from left to right:

(a) \[ 1 + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} \]
(b) \[ 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 10^{-10} + 1 \]

You should get the answer 1.000000000 in (a) and 1.000000001 in (b). The reason of getting an incorrect answer in (a) is that when \( 10^{-10} \) is added to 1, it is too small and is truncated. The rearrangement in (b) provides a correct result.

In conclusion, as we mentioned before, the way of rearranging a formula to reduce computational errors depends very much on the particular characteristics and nature of the problems themselves. We have no intention, and also not possible, to exhaust them all. Readers would identify how to rewrite a formula to reduce computational errors of a particular problem by experience.
**SUMERROR1** 6-FEB-1986 10:52

1! SUMERROR1
2! VAX 11/7501 VAX-11 BASIC V2.3
3! BY GLORY PONG

10! **PROGRAM DEMONSTRATES Rounding ERROR**
11! **SUMMING \( \frac{1}{N} \) N TIMES**
12! **USING SINGLE PRECISION**
13!

100! **DECLARE SINGLE PRECISION**
110 DECLARE SINGLE SUM
120 DECLARE INTEGER CONSTANT N=100
140 SUM = 0.

200!
201! **LOOP TO SUM \( \frac{1}{N} \)**
210 FOR I = 1 TO N
220 SUM = SUM + 1. / N
230 NEXT I

400!
401! **PRINT RESULT**
410 PRINT "SUMMING \( \frac{1}{N} \) IS EQUAL TO";SUM
999 END

**READY**

**RUN**
SUMERROR1 6-FEB-1986 10:53

SUMMING \( \frac{1}{N} \) IS EQUAL TO .999999
**READY**
**APPENDIX II**

SUMERROR2 6-FEB-1986 10:55

1! SUMERROR2
2! VAX 11/750; VAX-11 BASIC V2.3
3! BY GLORY PONG
10!
11! **PROGRAM DEMONSTRATES ROUNING ERROR
12! **SUMMING (1/N) N TIMES
13! **USING DOUBLE PRECISION
100!
101! **DECLARE DOUBLE PRECISION
110 DECLARE DOUBLE SUM
120 DECLARE INTEGER CONSTANT N=100
140 SUM = 0.
200!
201! **LOOP TO SUM (1/N)
210 FOR I = 1 TO N
220 SUM = SUM + 1. / N
230 NEXT I
400!
401! **PRINT RESULT
410 PRINT "SUMMING (1/N) IS EQUAL TO";SUM
999 END

Ready

RUN
SUMERROR2 6-FEB-1986 10:55

SUMMING (1/N) IS EQUAL TO 1
Ready
**PROGRAM DEMONSTRATES ROUNдинG Error**
**TO SOLVE THE FOLLOWING SIMULTANEOUS EQUATIONS BY**
**CRAMER’S RULE**:

14! ** 0.2038X + 0.1218Y = 0.2014
15! ** 0.4071X + 0.2436Y = 0.4038
16! ** THE EXACT SOLUTION SHOULD BE X=-2; Y=5

**DEFINE VARIABLES**
110 DECLARE SINGLE A,B,C
120 DECLARE SINGLE D,E,F
130 DECLARE SINGLE X,Y

**ASSIGN VALUES**
210 A = 0.2038
220 B = 0.1218
230 C = 0.2014
240 D = 0.4071
250 E = 0.2436
260 F = 0.4038

**THE CRAMER’S RULE**
310 X = (C*E - B*F) / (A*E - B*D)
320 Y = (A*F - C*D) / (A*E - B*D)

**PRINT THE RESULT**
410 PRINT ’X = ’; X, ’Y = ’; Y

```
RUN
CRAMERR1 6-FEB-1986 10:57
X = -2.00012  Y = 5.00031
Ready
```
**PROGRAM DEMONSTRATES ROUNding ERROR**
**TO SOLVE THE FOLLOWING simultaneous equations by**
**Cramer's RULE**
**THE EXACT solution should be x=-2; y=5**

**DEFine VARIABLES**
**ASSign VALUES**
**THE CRAMER'S RULE**
**PRINT The RESULT**

```
RUN
CRAMERR2  6-FEB-1986 10:59
X = -2.00006  Y = 5.0001
Ready
```

```
APPENDIX IV

CRAMERR2  6-FEB-1986 10:58

1!  CRAMERR2
2!  VAX 11/7501 VAX-11 BASIC V2.3
3!  BY GLORY PONG
10!
11!  ***PROGRAM DEMONSTRATES ROUNding ERROR
12!  ***TO SOLVE THE FOLLOWING simultaneous equations by
13!  ***Cramer's RULE
14!  ***  0.2038X + 0.1218Y = 0.2014
15!  ***  0.4071X + 0.2436Y = 0.4038
16!  ***THE EXACT solution should be X=-2; Y=5
100!
101!  **DEFINE VARIABLES
110 DECLARE DOUBLE A,B,C
120 DECLARE DOUBLE D,E,F
130 DECLARE DOUBLE X,Y
200!
201!  **ASSIGN VALUES
210 A = 0.2038
220 B = 0.1218
230 C = 0.2014
240 D = 0.4071
250 E = 0.2436
260 F = 0.4038
300!
301!  **THE CRAMER'S RULE
310 X = (C*E - B*F) / (A*E - B*D)
320 Y = (A*F - C*D) / (A*E - B*D)
400!
401!  **PRINT THE RESULT
410 PRINT 'X = ',X,'; Y = ',Y
999 END

Ready
```
**APPENDIX V**

**CRAMERR4  6-FEB-1986 11:03**

1!  CRAMERR2
2!  VAX 11/750; VAX-11 BASIC V2.3
3!  BY GLORY PONG
10!
11!  **PROGRAM DEMONSTRATES ROUNCNG ERROR**
12!  **TO SOLVE THE FOLLOWING SIMULTANEOUS EQUATIONS BY**
13!  **CRAMER'S RULE :**
14!  **0.2038X + 0.1218Y = 0.2014**
15!  **0.4071X + 0.2436Y = 0.4038**
16!  **THE EXACT SOLUTION SHOULD BE X=-2; Y=5**
20!
21!  **THE TWO EQUATIONS ARE MULTIPLIED BY 10000**
22!  **THE SOLUTION SET IS THE SAME**
100!
101!  **DEFINE VARIABLES**
110 DECLARE SINGLE A,B,C
120 DECLARE SINGLE D,E,F
130 DECLARE SINGLE X,Y
200!
201!  **ASSIGN VALUES**
210 A=2038
220 B=1218
230 C=2014
240 D=4071
250 E=2436
260 F=4038
300!
301!  **THE CRAMER'S RULE**
310 X = (C*E - B*F) / (A*E - B*D)
320 Y = (A*F - C*D) / (A*E - B*D)
400!
401!  **PRINT THE RESULT**
410 PRINT 'X = ',X,'; Y = ',Y
999 END

Ready

RUN
CRAMERR4  6-FEB-1986 11:04

X = -2
Y = 5
Ready
APPENDIX VI

BINOMCO1 6-FEB-1986 11:18

1!  BINOMCO1
2!  VAX 11/750; VAX-11 BASIC V2.3
3!  BY  GLORY PONG
10!
11!  **PROGRAM FINDS BINOMIAL COEFFICIENT
12!  **BY EVALUATING ALL THE FACTORIALS FIRST
100!
101!  **DEFINE VARIABLES
110  DECLARE INTEGER BOUND,N,R
120  DECLARE REAL FACT,NFACT,RFACT
140!
141!  **INPUT VALUES
150  INPUT 'PLEASE GIVE THE VALUE OF N',N
160  INPUT 'PLEASE GIVE THE VALUE OF R',R
200!
201!  **EVALUATE N FACTORIAL
210  BOUND = N
220  GOSUB 1000
230  NFACT = FACT
300!
301!  **EVALUATE R FACTORIAL
310  BOUND = R
320  GOSUB 1000
330  RFACT = FACT
400!
401!  **EVALUATE (N-R) FACTORIAL
410  BOUND = N - R
420  GOSUB 1000
500!
501!  **CALCULATE THE COEFFICIENT
510  COEF = NFACT / (RFACT * FACT)
600!
601!  **PRINT THE COEFFICIENT.
610  PRINT 'THE COEFFICIENT IS ';COEF
620  PRINT
999  GOTO 150
1000!
1001!  **SUBROUTINE TO EVALUATE FACTORIAL
1010  FACT = 1
1020  IF BOUND < 2 THEN RETURN
1030  FOR I = 2 TO BOUND
1040  FACT = FACT * I
1050  NEXT I
1060  RETURN
1070!
9999  END

Ready

RUN
BINOMCO1 6-FEB-1986 11:19

PLEASE GIVE THE VALUE OF N ? 34
PLEASE GIVE THE VALUE OF R ? 3
%BAS-F-FLOPOIERR, Floating point error or overflow
-BAS-I-USEFC_PSL, at user PC=001481B0, PSL=03C00020
-BAS-I-FROLINGSB, from line 1040 in GOSUB 1000 in module BINOMCO1
-BAS-I-FROLINMOD, from line 220 in module BINOMCO1
-SYSTEM-F-FLTOVF_F, arithmetic fault, floating overflow at PC=0005B9!
**PROGRAM FINDS BINOMIAL COEFFICIENT BY THE MODIFIED FORMULA**

100 DEFINE VARIABLES
110 DECLARE INTEGER N, R
120 DECLARE REAL COEF
141 **INPUT VALUES**
150 INPUT 'PLEASE GIVE THE VALUE OF N', N
160 INPUT 'PLEASE GIVE THE VALUE OF R', R
200 **EVALUATE THE COEFFICIENT**
205 COEF = 1
210 IF R = 0 THEN 300
220 FOR I = 0 TO R - 1
230 COEF = COEF * (N - I) / (R - I)
240 NEXT I
300 **PRINT THE COEFFICIENT**
310 PRINT 'THE COEFFICIENT IS ', COEF
320 PRINT
330 END

RUN
BINOMC02 6-FEB-1986 11:22

PLEASE GIVE THE VALUE OF N ? 34
PLEASE GIVE THE VALUE OF R ? 3
THE COEFFICIENT IS 5934

Ready
Print of lesson HELLO

g:yes
d:n$(30)
d:n1$(8)
d:et$(12)
d:n2$(2)
t$:=t1;t1
t$:=g10,7
t:Welcome to Lessons on
t$:=u2;t1;y2,14
t:Numerical Analysis
t$:=s15,50;12,50;8,50;15,50;12,50;8,50
t$:=s13,50;17,70;13,90;17,110;15,150
w:4
g:es;v3,39,4,23
t$:=s1;t1
t:I am the Apple computer.
t:
t:We communicate through the screen and the keyboard.
t:
t:Be sure to hit the RETURN key every time after you finish typing.
t:
t:Let's get better acquainted first.
*t:ame
't$:=g9,13
th:What is your name?
a:$m$
j(len(m)=0):name
r:***************
r:* Link to Error *
r:***************
l:error
This lesson talks about errors that may be occurred in Numerical Analysis. It should take you about 30 minutes to complete this lesson. At any time if you want to study the previous topic again, please type @. Are you ready to start?

Before we look into how errors would be occurred, it is better to understand the representation of numbers inside a computer, like me! Numbers are usually represented as:

- Sign
- Exponent
- Mantissa

Hence +04 .123456

Represents 0.123456 × 10 or 1234.56

The number could be stored in:
- a) binary form;
- b) packed decimal form;
- c) hexadecimal form; etc.
Detail of these can be found in most elementary numerical analysis books.

However, no matter how numbers are stored -- base 2, base 10, or base 16; all stored numbers have to have only finite number of digits. Some may have 8 digits. Some may have 10 digits. I have 13. Some may have even more.

Anyway, as long as only finite number of digits is possible, error is expected to occur in representing some numerical values. For example, irrational numbers such as:

\[ 2 = 1.414213562\ldots \]
\[ \sqrt{2} \approx 1.414213562 \]
\[ \pi \approx 3.141592654 \]
\[ e \approx 2.718281828 \]

Or some fractions, such as:

\[ \frac{1}{3} = 0.333333333\ldots \]

cannot be expressed in finite number of digits.

There are other possible sources of errors. In general, they can be induced in three ways:

1) Error due to the nature of data and data transfer.
2) Error due to truncation and rounding.
3) Error due to inappropriate interpretation of formulas.

We are going to discuss them in detail.
Error of these types can only be corrected by analysing the obtained data using statistical techniques or by repeating the experiment or observation again.

Sometimes error may be induced when data are transferred, or copied. This type of error can be minimized by:

a) double check;
b) parity check;
c) range check;
d) consistency check;
e) check digit; etc.

It is sometimes error may be induced when data are transferred, or copied. This type of error can be minimized by:

a) double check;
b) parity check;
c) range check;
d) consistency check;
e) check digit; etc.

Link to ROUND
Press space bar to continue »»

j(len(%b)=0):@p
j:pto
p:1e
g:es
t:1
:e:

*sysx
l:hello
The second type of general error is rounding and truncation error.

The first principle of calculus gives

\[ \frac{dy}{dx} = \frac{f(x+h) - f(x)}{h} \]

as \( h \) approaches zero.

Let us see how \( f'(x) \) changes as \( h \) gets smaller by the formula using \( \exp(x) \) as example.

\[ f(x) = \frac{\exp(x)}{x}; \quad x = 20 \]

\[ \begin{array}{ll}
  h & f'(20) \quad \text{Error} \\
  0.01 & \frac{5.045346.78}{2.3045346.07} \\
  0.001 & \frac{5.045346.78}{2.3045346.07} \\
  0.0001 & \frac{5.045346.78}{2.3045346.07} \\
  0.00001 & \frac{5.045346.78}{2.3045346.07} \\
\end{array} \]

As you can see, the error is improving first, but getting worse as \( h \) is too small.

Try it yourself.
Let us look at another simple example.

This example sums $1/n$ $n$ times for any positive integer $n$. It is too obvious that the result should be 1. However, a computer like me, cannot always get the exact result without error.

Let me demonstrate how poor my arithmetic is ....

I am very slow, please be patient!

Let's try some numbers yourself. Remember, I am very slow, so don't give me any value larger than 50 (0 to exit).

```
As you have seen, I may have error in seven digits after decimal.

In fact, if I sum $(1/67)$ 67 times, I only get $0.999999$ with an error 5.96046E-07.

It is true that the larger number I use, the worse I get.

So much for this. For more examples, please refer to the notes and the appendices.

As the conclusion, rounding error and truncation error can affect the result no matter how simple the calculation is.

We should always use our common sense and knowledge to determine whether the numerical result is acceptable or not.
\[ f = e = f - fl \]

*sysx

l: error
Print of lesson FORMUL

The third type of general error is error due to inappropriate interpretation of formulas, this depends very much on the nature of the problem itself, it is not possible to exhaust all cases.

We only bring out some examples as guidelines.

Sometimes we may come across the factorials, such as, the combination of n objects, taken r at a time, is given by the formula

\[ C_{n}^{r} = \frac{n!}{r!(n-r)!} \]

If we evaluate all the factorials first, then the multiplications and divisions, as the way the formula stated, we may easily get overflow.

If the formula is rearranged to

\[ C_{n}^{r} = \frac{n(n-1)(n-2)...(n-r+1)}{r(r-1)(r-2)...1} \]

not only the overflow problem can be resolved, the number of calculations also reduced, hence rounding error would be improved.

For computational examples, please refer to appendices VI & VII in the lesson notes.

The following two problems again demonstrates the rearranging of calculations can give different results.

Use a 10-digit machine to compute
Every body knows that ten billion plus one is equal to ten billion and one. However, in a 10-digit computer, ten billion is stored as 1.000000000E10. If one is added to it, 1 has to be normalized to the same power E10. The answer is still ten billion, in 10-digit-mantissa. Therefore in problem (a), no matter how many ones are added to the ten billion, the answer is still ten billion. But in problem (b), if the ones are summed first, the cumulated result can be sufficiently large to be added into the ten billion.
be over exaggerated, it actually happens very often in numerical problems.

Let me show you one last example.

Sometimes we may need to evaluate high power polynomials, such as

\[ P(x) = 12x + 10x + 8x + 6x + 4x + 2x + 1 \]

For large values of \( x \), the mentioned problem may happen again that the last few terms would not be added into it because they are comparatively too small.

In order to overcome this, a factorization method, called the Horner's rule, can rearrange a polynomial into a way that not only the error mentioned can be minimized, it also reduces the number of calculations involved.

Let me show you the Horner's rule.

Let \( P(x) \) be

\[ P(x) = 6x^5 + 5x^4 + 4x^3 + 3x^2 + 2x + 1 \]

Just consider the first two terms, they have a common factor of 5.

\[ (12x + 10)x + 8x + 6x + 4x + 2x + 1 \]
Then the next term, 
with the new $x$ term, 
they have a common factor of $x$,

set

$(x + 8)$

$(x)$

$6x + ...$
This process goes on until only the constant left.

Do you think you can do it now?
As you can see, the Horner's rule can rearrange a polynomial in a way that not only large values of \( x \) is taken care of, it also reduces the number of calculations, which also means less computational error and computer time.

As the conclusion, the way of rearranging a formula to reduce computational errors depends very much on the particular characteristics and nature of the problems themselves. We have no intention, and also not possible, to exhaust them all. Sometimes trial and error is needed. And past experience helps.

This conclude the lesson on Error:

See you soon, $m$! The End
Print of lesson END

gives
*end
tsg0,8
t: Please take out the disk, turn off the computer and answer the questions.
tsg0,22
as:
j: end
HONG KONG POLYTECHNIC

DEPARTMENT OF MATHEMATICAL STUDIES

QUESTIONS ON ASSIGNMENT PROBLEM 1

Class: __________ Date: ______________

Name: ______________________________

Please do not read the questions before you have completed the package and do not rerun the package while answering the questions.

Section A

1. The reason for a stored number may not be the exact value is because

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

2. Give four methods to minimize error in data transfer.

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

3. In the example of first principle of calculus, what h value did you try? How does the error compare with others?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________
4. In the example of summing \( \frac{1}{n} \) \( n \) times, what values of \( n \) does the computer use? Which have error zero?

5. Factorize the following using Horner's Rule:
\[ x^4 - 3x^2 + 2x - 4 \]

Section B

1. How long did you take to finish the package? \[ \text{min.} \]

2. Can you type? If you can, how fast?

3. What computer experience do you have?

4. Do you have a personal computer?

5. Have you taken any similar package like this one before?

If you do, what package?
Section C

Please use the following scale to circle your answers.

6 - strongly agree, 1 - strongly disagree

1. The procedure for using the package is simple. 6 5 4 3 2 1
2. The length of this package is about right. 6 5 4 3 2 1
3. The screens are easy to read. 6 5 4 3 2 1
4. Interaction between user and computer is adequate. 6 5 4 3 2 1
5. The content of this package is well presented. 6 5 4 3 2 1
6. The examples are easy to follow. 6 5 4 3 2 1
7. Accompanying written lesson notes are helpful. 6 5 4 3 2 1
8. A CAL package like this in Numerical Analysis is appropriate. 6 5 4 3 2 1
9. If time allows, I would like to have more computer packages available. 6 5 4 3 2 1
10. I would like to borrow a package and use it at home or in the office. 6 5 4 3 2 1

11. Please describe briefly anything you particularly like about this package.

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

12. Please describe briefly anything you particularly dislike about this package.

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________
Section D

Please comment in not less than 200 words about computer aided learning. You could include the advantages, disadvantages, suitability in Hong Kong environment, etc. of using computer aided learning.
Topic: Numerical Integration

Keywords

Area under a curve, quadrature, trapezoidal rule, Simpson's rule, truncation errors, Newton-Cote methods, Gauss quadrature, Romberg integration, improper integrals, piecewise continuous, multiple integrals.

Objectives

After working through this section, you should be able to

(i) find the area under a curve applying trapezoidal rule and Simpson's rule

(ii) evaluate the truncation errors of these two methods

(iii) find the area under a curve applying Romberg integration and Gauss quadrature

(iv) distinguish the differences between area under a curve and definite integral

(v) understand the accuracy and limitations of the four algorithms for numerical integration
Learning Activity

1. Introduction

Very often we would like to evaluate a definite integral which may not be easy, or even possible. For example, we want to find the arch length of $\sin x$, between $0$ and $\pi$, which can be written as

$$\int_0^\pi \sqrt{1 + \cos^2 x} \, dx$$

However, even this straightforward definite integral problem cannot be solved. Fortunately, in most cases, definite integral can be considered as the Riemann sum. Therefore

$$\int_a^b f(x) \, dx$$

may be regarded as to find the area of $f(x)$ between $[a, b]$. And the definite integral approximated by finding the area under the curve is known as numerical integration, or quadrature.

The terms numerical integration and area under a curve sometimes are also used interchangeably. In fact, they are the same if $f(x)$ does not cut the x-axis within the interval. However, if $f(x)$ cuts the x-axis, some $f(x)$ values may cancel one another due to the change of signs of $f(x)$ along $x$. As a rule of thumb, in finding area under $f(x)$, all evaluated $f(x)$ values should take the absolute values to avoid undesirable cancellation.

Although numerical integration seems only appropriate on definite integral, it is sometimes also useful in investigating the behavior of the indefinite integral over a certain interval.

There are a number of methods used to find the area under a curve:

(a) Counting squares - plot the curve on a graph paper as accurate as possible, then count the number of squares covered by the curve. Sometimes, instead of counting squares, we would cut the covered part and find the mass of it, then proportionate the mass with the area.

(b) Simulation - generate numerous random points within a convenient interval which covered more than the area in question. The proportion of points that fall within the curve is that of the area.
(c) Rectangular rule - the curve is being divided into \( n \) equal-width rectangles. The height of the rectangle can be either the function in the left side or that in the right side of the rectangle. The total area is the sum of the areas of the \( n \) rectangles. It is natural to think that the more rectangles we use, the more accurate we should get. However, we have to bear in mind that there is a trade-off between accuracy and time needed in evaluating the area of the rectangles. Also, when the width of the rectangle becomes very small, the rounding error may be large too.

There are some other common algorithms in finding the area under the curve. We shall discuss them in more detail.

2. Trapezoidal Rule

In trapezoidal rule, we divide \([a, b]\) into \( n \) equal parts, that is \( a = x_0, x_1, \ldots, x_n = b \), with width \( h = (b - a)/n \). Each interval is approximated as a trapezium, hence, the area of the first one is

\[
A_1 = h\left[f(x_0) + f(x_1)\right]/2
\]

The area of the second one is

\[
A_2 = h\left[f(x_1) + f(x_2)\right]/2
\]

And the total area \( A \) is

\[
A = A_1 + A_2 + A_3 + \ldots + A_n = h\left[f(x_0) + 2f(x_1) + 2f(x_2) + \ldots + 2f(x_{n-1}) + f(x_n)\right]/2
\]

Or

\[
A = h\left[f(x_0)/2 + f(x_1) + f(x_2) + \ldots + f(x_{n-1}) + f(x_n)/2\right]
\]

Example (1)

Use the Trapezoidal rule to approximate the area under the curve \( f(x) = x^2 \) for \( x \) from 0 to 2 using 8 intervals.

Since \( h = (2 - 0)/8 = .25 \)

<table>
<thead>
<tr>
<th>( x )</th>
<th>0</th>
<th>.25</th>
<th>.5</th>
<th>.75</th>
<th>1</th>
<th>1.25</th>
<th>1.5</th>
<th>1.75</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f(x) )</td>
<td>0</td>
<td>.0625</td>
<td>.25</td>
<td>.5625</td>
<td>1</td>
<td>1.5625</td>
<td>2.25</td>
<td>3.0625</td>
<td>4</td>
</tr>
</tbody>
</table>
Therefore Area = \((.25)(0/2 + .0625 + .25 + .5625 + 1 + 1.5625 + 2.25 + 3.0625 + 4/2)\)
\[ = (.25)(10.75) \]
\[ = 2.8275 \text{ square units.} \]

In fact, trapezoidal rule is an algorithm of approximating a function by n continuous straight lines. This method, compared with the rectangular rule, which also divides the interval into n-equal parts, but the trapizoidal rule by taking only one more function evaluation, improves the result significantly.

3. Error of the Trapezoidal Rule

We are going to discuss what the error is in finding the area by trapezoidal rule.

From Newton-Gregory forward difference interpolation formula we have

\[ y = f_p \]
\[ = f_0 + p\Delta f_0 + \frac{p(p-1)}{2!} \Delta^2 f_0 + \frac{p(p-1)(p-2)}{3!} \Delta^3 f_0 + ... \]

Since \( x = x_0 + ph \), \( dx = hdp \)

Therefore \( \int ydx = \int y \frac{dx}{dp} dp \)
\[ = h \int ydp \]

The area under the curve between the points \( x_0 \) and \( x_1 \) with respect to \( p \) is

\[ h \int_{0}^{1} ydp \]
\[ = h \int_{0}^{1} \left( f_0 + p\Delta f_0 + \frac{(p^2-p)}{2!} \Delta^2 f_0 + \frac{p^3-3p^2+p}{3!} \Delta^3 f_0 + ... \right) dp \]
\[ = h \left[ pf_0 + \frac{p^2\Delta f_0}{2} + \frac{(p^3 - p^2)}{6} \Delta^2 f_0 + \frac{(p^4 - p^3)}{24} \Delta^3 f_0 + ... \right]_0^1 \]
\[ = h \left[ f_0 + \frac{\Delta f_0}{2} - \frac{1}{12} \Delta^2 f_0 + \frac{1}{24} \Delta^3 f_0 - ... \right] \]
If the second and higher orders are ignored, the area becomes

\[ h \left( f_0 + \frac{\Delta f_0}{2} \right) \]
\[ = h \left[ f_0 + \frac{(f_1 - f_0)}{2} \right] \]
\[ = h\frac{f_0 + f_1}{2} \]

which is exactly the area of the trapezium. Therefore the maximum error using the trapezoidal rule for one trapezium is \( h^2 f_0 / 12 \).

However, from calculus, we know that

\[ f' = \lim_{h \to 0} \frac{\Delta f}{h} \]

Therefore, approximately we can define that \( \Delta f = hf' \) for small \( h \).

Similarly, we can claim the approximations of \( \Delta^2 f = h^2 f'' \).

Therefore the maximum error for trapezoidal rule is \( h^3 f''_0 / 12 \).

If we let \( |f''(x)| \leq M \), for some \( M \) and for all \( x \) in the range \([a, b]\), then the error \( E' \) is

\[ E' \leq h^3 M / 12 \]

If there are \( n \) trapeziums to be added to give the total area under the curve in \([a, b]\), then the total error \( E \) is

\[ E \leq nh^3 M / 12 \]
\[ = nh \cdot h^2 M / 12 \]
\[ = (b - a) h^2 M / 12 \]

Since \((b - a)\) and \( M \) are constants, therefore we may claim that, ignoring the rounding error, the truncation error of the trapezoidal rule is directly proportional to the square of the width. That is, if we reduce the width by a factor of 2, the error should be decreased by a factor of 4.
4. Simpson's Rule

Since the trapezoidal rule approximates the function by \( n \) continuous straight lines. Therefore we might think that an algorithm that approximates the function by quadratic lines, or cubic lines, etc., might yield a better approximation and, hopefully not too complicated. We are going to discuss the Simpson's rule which approximates the area of a function by successive parabolic lines.

The equation of the parabola, in general, can be given as

\[
f(x) = ax^2 + bx + c
\]

where \( a, b, c \) are constants and to be determined. Since three points are needed to determine a parabola, and without loss of generality, we may assume that these points be \((-h, y_0), (0, y_1), (h, y_2)\). Substituting these points into the equation, we have

\[
\begin{align*}
y_0 &= ah^2 - bh + c \\
y_1 &= c \\
y_2 &= ah^2 + bh + c
\end{align*}
\]

Combining, \( ah^2 - bh = y_0 - y_1 \)

\[
ah^2 + bh = y_2 - y_1
\]

Implies \( 2ah^2 = y_0 + y_2 - 2y_1 \)

However, from integration, the exact area \( A \), of a parabola is

\[
A = \int_{-h}^{h} f(x) \, dx = \int_{-h}^{h} (ax^2 + bx + c) \, dx
\]

\[
= \left[ \frac{ax^3}{3} + \frac{bx^2}{2} + cx \right]_{-h}^{h}
\]

\[
= \frac{2ah^3}{3} + 2ch
\]

\[
= h(2ah^2 + 6c)/3
\]

To get rid of \( a \), substituting from above, we have

\[
A = \frac{h(y_0 + y_2 - 2y_1 + 6y_1)}{3}
\]

\[
= \frac{h(y_0 + 4y_1 + y_2)}{3}
\]
Since we are using two equal parts for each parabola, therefore, for \( n \) parabolas, we need \( 2n \) equal parts, and the total area is given as

\[
h[(y_0 + 4y_1 + y_2) + (y_2 + 4y_3 + y_4) + \ldots]/3
\]

\[
= h[y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \ldots + y_n]/3
\]

\[
= h[(y_0 + y_n + 4(y_1 + y_3 + \ldots + y_{n-1}) + 2(y_2 + y_4 + \ldots + y_{n-2})]/3
\]

Example (2)

Use the Simpson's rule to approximate the area under the curve \( f(x) = x^2 \) for \( x \) from 0 to 2 using 4 intervals.

Since \( h = (2 - 0)/4 = .5 \)

\[
\begin{array}{|c|c|c|c|c|}
\hline
x & 0 & .5 & 1 & 1.5 & 2 \\
\hline
f(x) & 0 & .25 & 1 & 2.25 & 4 \\
\hline
\end{array}
\]

Therefore area \( = (.5)[0 + 4 + 4(.25 + 2.25) + 2(1)]/3 \)

\[
= (.5)(16)/3
\]

\( = 8/3 \) square units.

The actual area, from calculus is

\[
\int_0^2 x^2 dx = \frac{2}{3} \left[ \frac{3}{2} \right]_0^2
\]

\( = 8/3 \).

The reason we can obtain the exact value is because Simpson's rule is derived from the equation of parabola, and we are using that to find the area of \( f(x) = x^2 \), which is a quadratic, therefore we should expect an accurate result. However, if \( f(x) \) is of higher degree, error would occur.

5. Error of the Simpson's Rule

The deduction of the truncation error of the Simpson's rule is similar to that of the trapezoidal rule and is left as an exercise. Since the maximum error \( E \) is \( (b - a)h^4 M/180 \); where \( M \leq |f'''| \). it is clear that the error is directly proportional to \( h^4 \). Hence the error decreases much faster as more intervals are used.
DIM DIFFER(20)
DEF FNF(X)-X*EXP(-X) ' Define function

SCREEN 1,1 ' Medium resolution graphics
DELAY 1
LOCATE 10,8,0
PRINT "Numerical Integration"
LOCATE 24,6,0
CLEARKEYS-INKEYS$ PRINT "Press SPACE BAR to continue";
WHILE NOT INSTAT : WEND
CLEARKEYS-INKEYS$

SCREEN 2,1 ' High resolution graphics
DELAY 1
LOCATE 3,35,0
PRINT "Introduction" ' Heading
LINE (267,14) - (370,25),B ' Box
LOCATE 5,1,0
PRINT " Very often we would like to evaluate a definite integral which"
PRINT " may not be easy, or even possible. For example, the function"
PRINT " f(x) = sin x , between 0 and 180 degrees, is as follow."
GOSUB PRESSBAR

VIEW (10,80) - (300,180),6,1 'Define graphic window
WINDOW (-.2,-.2) - (3.3,1.6)
LINE (.1,0) - (3.2,0) ' X-axis
LINE (0,1.5) - (0,-.1) ' Y-axis
FOR X=0 TO 3.14 STEP .01
  Y=SIN(X)
  PSET(X,Y)
NEXT X
GOSUB PRESSBAR

LOCATE 8,1,0
PRINT " The arch length is the integral of g(x) = [1 + (cos x)^2],"
VIEW (0,0) - (599,69)
WINDOW (0,0) - (599,69)
LINE (378,10) - (382,7) ' Draw square root sign
LINE (382,7) - (386,14)
LOCATE 9,1,0
PRINT " which cannot be solved."
GOSUB PRESSBAR

VIEW (320,80) - (600,180),6,1 ' Second window
WINDOW (-.2,-.2) - (3.3,1.6)
LINE (.1,0) - (3.2,0)
LINE (0,1.5) - (0,-.1)
FOR X=0 TO 3.14 STEP .01
    Y=SQR(1+(COS(X)^2))
    PSET(X,Y)
NEXT X
LINE (3.14,0) - (3.14,Y)
GOSUB PRESSBAR

VIEW (5,26) - (639,75)
CLS
LOCATE 5,1
PRINT "Fortunately, most definite integral can be considered as the"
PRINT "Reimann sum, hence can be approximated by finding the area"
PRINT "under the curve."
GOSUB PRESSBAR

VIEW (320,80) - (600,180)
WINDOW (-.2,-.2) - (3.3,1.6)
FOR X=0 TO 3.14 STEP .01
    Y=SQR(1+(COS(X)^2))
    LINE (X,0) - (X,Y)
NEXT X
GOSUB PRESSBAR

VIEW (5,26) - (639,199)
CLS
LOCATE 5,1
PRINT "However, the value for definite integral may be negative, while"
PRINT "the area must be positive. Therefore, when finding the numerical"
PRINT "integral, the sign should be reserved.";
GOSUB PRESSBAR

VIEW (40,80) - (560,180),6,1
WINDOW (-.4,-.4) - (6.4,1.4)
LINE (-.1,0) - (6.3,0)
LINE (0,1.1) - (0,-1.1)
LINE (1.8,0) - (1.8,SIN(1.8))
LINE (6,0) - (6,SIN(6))
FOR X=1.8 TO 6 STEP .01
    Y=SIN(X)
    PSET (X,Y)
NEXT X
LOCATE 14,45,1
PRINT "Numerical integration"
FOR X=1.8 TO 6 STEP .01
    Y=SIN(X)
    LINE (X,0) - (X,Y)
NEXT X
GOSUB PRESSBAR

CLS
LOCATE 7,44,0
PRINT "And when finding the"
PRINT " area under a curve, absolute value should be used."
LINE (-.1,0) - (6.3,0)
LINE (0,.1) - (0,-.1)
LINE (1.8,0) - (1.8,SIN(1.8))
LINE (6,0) - (6,SIN(6))
FOR X=1.8 TO 6 STEP .01
  Y=SIN(X)
  PSET (X,Y)
NEXT X
PRINT "Area under the curve" FOR X=1.8 TO 6 STEP .01
  Y=SIN(X)
  LINE (X,0) - (X,ABS(Y))
NEXT X
GOSUB PRESSBAR

VIEW (5,26) - (639,199) CLEAR WINDOW
CLS
LOCATE 9,1,1
PRINT "We are going to investigate two algorithms in Numerical Integration"
PRINT
PRINT "1. The Trapezoidal Rule"
PRINT "2. The Simpson's Rule"
GOSUB PRESSBAR

SCREEN 1,1 LOW RESOLUTION GRAPHICS
DELAY 1
LOCATE 10,10,0 PRINT "Trapezoidal Rule"
DELAY 4
SCREEN 2 HIGH RESOLUTION GRAPHICS
DELAY 1
LOCATE 3,30,0 PRINT "Trapezoidal Rule"
LINE (227,14) - (365,25),,B BOX
LOCATE 5,1,0 PRINT "The formula for Trapezoidal Rule is given as :"
PRINT "Area = h [f(a)/2 + f(x1) + f(x2) + ... + f(b)/2]"
PRINT "And the truncation error is given as :"
PRINT "2"
PRINT "Error = (b - a) h M / 12"
PRINT "Let's use this formula to find the numerical integration of the"
PRINT "following function and investigate the error terms :"
PRINT
PRINT " f(x) = xe" ; for x between 0 and 1"
GOSUB PRESSBAR

VIEW (0,26) - (639,199)
CLS
LOCATE 5,1,0
PRINT " -x"
PRINT " f(x) = xe" ; [0,1]
VIEW (315,56) - (635,156),6,1
WINDOW (-.1,-.1) - (1.2,.5)
LINE (.05,0) - (.1,0)
LINE (0,.45) - (0,-.03)
FOR X=0 TO 1 STEP .002
  Y=FNF(X)
  PSET (X,Y)
NEXT X
LINE (1,0) - (1,FNF(1))
LOCATE 19,42,1
PRINT "0"
LOCATE 19,74,1
PRINT "I"
GOSUB PRESSBAR

LOCATE 8,1,0
PRINT " Using n trapeziums"
VIEW (0,64) - (300,180)
WINDOW (0,64) - (300,204)
LINE (50,96) - (50,190)
LOCATE 11,3,0
PRINT "N"
LOCATE 11,11,0
PRINT "AREA"
LINE (0,174) - (240,174)
ACTUAL=2*EXP(-1)+1
AREA=FNF(0)/2 + FNF(1)/2
VIEW (315,56) - (635,156)
WINDOW (-.1,-.1) - (1.2,.5)
N=2
FOR I=2 TO 8
  AREA=0
  LOCATE 11+I,1,0
  PRINT N
  H=1/N
  FOR J=1 TO N-1
    AREA=AREA + FNF(J*H)
    LINE (J*H,0) - (J*H,FNF(J*H))
  NEXT J
  AREA=H*(AREA+AREA1)
  LOCATE 11+I,10,0
  PRINT AREA
  N=2*N
IF 1<i AND I<6 THEN DELAY 2
NEXT I
LOCATE 22,1,0
PRINT "The actual value is ";ACTUAL
GOSUB PRESSBAR

VIEW (0,26) - (639,199)
WINDOW (0,0) - (80,20.5)
CLS
LOCATE 5,1,0
PRINT "Since Error = Actual - Calculated"
PRINT "And Error is proportional to h square"
PRINT "Therefore; if N increases by 2 ; then Error decreases by 4"
GOSUB PRESSBAR

LOCATE 12,3,0
PRINT "N"
LOCATE 12,12,0
PRINT "Error"
LOCATE 12,41,0
PRINT "Error of n / Error of 2n"
LINE (0,12) - (71,12)
LINE (7,13) - (7,5)
LINE (37,13) - (37,5)
N1-2
GOSUB TRAPEvaluate
GOSUB PRESSBAR

GOSUB RATIO
GOSUB PRESSBAR

TRY-1
TRAPTRIAL:
VIEW (5,26) - (639,75)
CLS
LOCATE 5,1,0
PRINT "As we can see, the values are very near 4 in the last column."
PRINT "However, rounding error begins to dominate as N gets large."
PRINT INPUT "Try it yourself. Please input the starting N (1-9, 0 to quit)? ",N1
$ IF N1$="O" THEN GOTO ENDTTRAPTRIAL
GOSUB INPUTCHECK ' To check input
IF N1=0 THEN GOTO TRAPTRIAL ' To clear table
GOSUB CLEARTABLE ' To evaluate area
GOSUB TRAPEvaluate ' To calculate ratio
TRY=TRY+1
IF TRY < 5 THEN GOTO TRAPTRIAL ' Allow 4 tries
GOSUB PRESSBAR
As a Summary:

1. Formula for Trapezoidal rule is simple
2. It needs n+1 function evaluations
3. Error is proportional to \( h^2 \) for small N
4. Rounding error dominates as N becomes large.

Let us now study the Simpson's Rule.

The formula for Simpson's Rule is given as:

\[
\text{Area} = h \left( f(a) + f(b) + 4 \left[ f(x_1) + f(x_3) + \ldots \right] + 2 \left[ f(x_2) + f(x_4) + \ldots \right] \right) / 3
\]

And the truncation error is given as:

\[
\text{Error} = (b - a) h^4 M / 180
\]

Let's again use this formula to find the numerical integration of the same function and investigate the error terms:

\[
f(x) = xe^{-x} \quad \text{for } x \text{ between } 0 \text{ and } 1
\]
PRINT "To integrate f(x) = xe ; [0,1]"

PRINT "Plot the function"

FOR X=0 TO 1 STEP .002
Y=FNF(X)
PSET (X,Y)
NEXT X

LOCATE 21,2,1
PRINT "0"
LOCATE 21,33,1
PRINT "1"
GOSUB PRESSBAR

LOCATE 8,41,0
PRINT "Using even N intervals"

VIEY (315,60) - (635,160)
WINDOW (0,64) - (300,204)
LINE (50,80) - (50,180)

ACTUAL = 2*EXP(-1)+1
AREA = FNF(0) + FNF(1)
N=2
FOR I=1 TO 6
LOCATE 12+I,41,0
PRINT N
H=1/N
AREA2=0
AREA4=FNF(H)
LINE (H,0) - (H,FNF(H))
FOR J=2 TO N-2 STEP 2
AREA2=AREA2 + FNF(J*H)
AREA4=AREA4 + FNF((J+1)*H)
LINE (J*H,0) - (J*H,FNF(J*H))
LINE ((J+1)*H,0) - ((J+1)*H,FNF((J+1)*H))
NEXT J
AREA = H * (AREA1 + 4*AREA4 + 2*AREA2) / 3
LOCATE 12+I,50,0
PRINT AREA
N=2*N
IF 0<1 AND 1<5 THEN DELAY 2
NEXT I
LOCATE 21,41,0
PRINT "The actual value is ";ACTUAL
GOSUB PRESSBAR

VIEW (0,26) - (639,199)
WINDOW (0,0) - (80,20.5)
CLS
LOCATE 5,1,0
PRINT " Since Error = Actual - Calculated"
PRINT " And Error is proportional to fourth power of h"
PRINT " Therefore; if N increases by 2 ; then Error decreases by 16"
GOSUB PRESSBAR

LOCATE 12,3,0
PRINT "N"
LOCATE 12,12,0
PRINT "Error"
LOCATE 12,41,0
PRINT "Error of n / Error of 2n"
LINE (0,12) - (71,12)
LINE (7,13) - (7,5)
LINE (37,13) - (37,5)
NL=2
GOSUB SIMPEVALUATE
GOSUB PRESSBAR
GOSUB RATIO
GOSUB PRESSBAR

TRY-1

SIMPTRIAL:

VIEW (5,26) - (639,75)
CLS
LOCATE 5,1,0
PRINT " For small N, the values are near 16 in the last column."
PRINT " However, rounding error begins to dominate as N increases."
PRINT INPUT " Try it yourself. Please input the starting N (2-8, 0 to quit)? ",N1
$ IF N1$="0" THEN GOTO ENDSIMPTRIAL
GOSUB INPUTCHECK
IF N1=0 THEN GOTO SIMPTRIAL
IF INT(N1/2)*2<>N1 THEN
BEEP 2
GOTO SIMPTRIAL
END IF
GOSUB CLEARTABLE
GOSUB SIMPEVALUATE
GOSUB RATIO
TRY=TRY+1
IF TRY < 4 THEN GOTO SIMPTRIAL
GOSUB PRESSBAR
ENDSIMPTRIAL:
As a Summary:

1. Formula for Simpson's rule is not complicated
2. It also needs \( n+1 \) function evaluations
3. Error improves significantly
4. Error is proportional to fourth power of \( h \)
5. Rounding error dominates quickly.

Conclusion

Trapezoidal Rule and Simpson's Rule have the following similarities:

1. They are based on the Newton-Cote method.
2. They all use equal space width \( h \).
3. \( f(a) \) and \( f(b) \) are in the formulas.
4. The formulas need \( N+1 \) function evaluations.

However, the two formulas have the following differences:

1. Trapezoidal Rule
   1. It approximates a curve by continuous straight lines.
   2. Error is only proportional to \( h \) square.
PRINT "3. Hence in general, it converges to the solution slower."
GOSUB PRESSBAR

VIEW (0,26) - (639,199)
CLS
LOCATE 9,1,0
PRINT "11. Simpson's Rule"
PRINT "1. It approximates a curve by continuous quadratic lines."
PRINT "2. Error is proportional to fourth power of h ." 
PRINT "3. Hence in general, it converges to the solution faster."
PRINT "4. Rounding error dominates quickly as N increases."
GOSUB PRESSBAR

VIEW (0,26) - (639,199)
CLS
LOCATE 11,1,0
PRINT "This concludes the Numerical Integration lesson."
LOCATE 17,1,0
PRINT "Please take out the disk and turn off the computer."
PRINT "Do not forget to answer the question paper."

CLEARKEY$-INKEY$
WHILE NOT INSTAT : WEND
GOTO FINISH

PRESSBAR:
' Press Bar prompt
CLEARKEY$-INKEY$
LOCATE 24,26,0,1,2
PRINT "Press SPACE BAR to continue":
WHILE NOT INSTAT : WEND
CLEARKEY$-INKEY$
LOCATE 24,26,0
PRINT ""
RETURN

TRAPEZOIDAL RULE FOR I-N1 TO N1+5
N=N1
FOR I=N1 TO N1+5
AREA=0
LOCATE 13+I-N1+1,1,0
PRINT N
H=1/N
FOR J=1 TO N-1
    AREA=AREA + FNF(J*H)
NEXT J
AREA=H*(AREA+AREA1)
LOCATE 13+I-N1+1,10,0
DIFFER(I) = ACTUAL - AREA
PRINT DIFFER(I)
N = 2 * N
NEXT I
RETURN

SIMPEVALUATE:
' Find area by Simpson's Rule
N = N1
FOR I = N1 TO N1 + 5
LOCATE 13 + I - N1 + 1, 1, 0
PRINT N
H = 1 / N
AREA2 = 0
AREA4 = FNF(H)
FOR J = 2 TO N - 2 STEP 2
AREA2 = AREA2 + FNF(J * H)
AREA4 = AREA4 + FNF((J + 1) * H)
NEXT J
AREA = H * (AREA1 + 4 * AREA4 + 2 * AREA2) / 3
LOCATE 13 + I - N1 + 1, 10, 0
DIFFER(I) = ACTUAL - AREA
PRINT DIFFER(I)
N = 2 * N
NEXT I
RETURN

RATIO:
' Find the error ratio
FOR I = N1 TO N1 + 4
LOCATE 14 + I - N1 + 1, 40, 0
IF DIFFER(I + 1) <> 0 THEN PRINT DIFFER(I) / DIFFER(I + 1)
ELSE PRINT "--" 
NEXT I
RETURN

INPUTCHECK:
' Check input error
IF LEN(N1$) <> 1 THEN
SOUND 160, 18
SOUND 60, 18
N1 = 0
RETURN END IF
N1 = VAL(N1$)
IF N1 = 0 THEN
SOUND 180, 18
SOUND 80, 18
RETURN END IF
RETURN

CLEARTABLE:
' Clear table
FOR I = 1 TO 8
LOCATE 13 + I, 1, 0
PRINT " "
LOCATE 13+I,10,0
PRINT " 
LOCATE 13+I,40,0
PRINT " 
NEXT I
RETURN

FINISH:
END
Numerical Integration

Press SPACE BAR to continue
Introduction

Very often we would like to evaluate a definite integral which may not be easy, or even possible. For example, the function $f(x) = \sin x$, between 0 and 180 degrees, is as follows.
Introduction

Very often we would like to evaluate a definite integral which may not be easy, or even possible. For example, the function \( f(x) = \sin x \), between 0 and 180 degrees, is as follows.

The arch length is the integral of \( g(x) = \sqrt{1 + (\cos x)^2} \), which cannot be solved.

Press SPACE BAR to continue
Introduction

Fortunately, most definite integrals can be considered as the Reimann sum, hence can be approximated by finding the area under the curve.
Introduction

However, the value for definite integral may be negative, while the area must be positive. Therefore, when finding the numerical integral, the sign should be reserved.

Press SPACE BAR to continue
Introduction

However, the value for definite integral may be negative, while the area must be positive. Therefore, when finding the numerical integral, the sign should be reversed. And when finding the area under a curve, absolute value should be used.

Press SPACE BAR to continue
Introduction

We are going to investigate two algorithms in Numerical Integration

1. The Trapezoidal Rule
2. The Simpson's Rule

Press SPACE BAR to continue
Trapezoidal Rule
The formula for Trapezoidal Rule is given as:

\[ \text{Area} = h \left[ \frac{f(a)}{2} + f(x_1) + f(x_2) + \ldots + \frac{f(b)}{2} \right] \]

And the truncation error is given as:

\[ \text{Error} = (b - a) h^2 M / 12 \]

Let's use this formula to find the numerical integration of the following function and investigate the error terms:

\[ f(x) = x e^{-x} \quad ; \quad \text{for } x \text{ between } 0 \text{ and } 1 \]

Press SPACE BAR to continue
To integrate $f(x) = xe^{-x}$ over $[0,1]$, use the Trapezoidal Rule.

Press SPACE BAR to continue.
The Trapezoidal Rule

To integrate \( \int_{0}^{1} xe^{-x} \, dx \)

Using \( n \) trapeziums

<table>
<thead>
<tr>
<th>( N )</th>
<th>AREA</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.243602529168129</td>
</tr>
<tr>
<td>4</td>
<td>0.2590450346469879</td>
</tr>
</tbody>
</table>
### Trapezoidal Rule

To integrate \( f(x) = x e^{-x} \); \([0,1]\)

Using \( n \) trapeziums

<table>
<thead>
<tr>
<th>( N )</th>
<th>( \text{AREA} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.243602529168129</td>
</tr>
<tr>
<td>4</td>
<td>0.2590450346469879</td>
</tr>
<tr>
<td>8</td>
<td>0.2629398107528687</td>
</tr>
<tr>
<td>16</td>
<td>0.2639156579971313</td>
</tr>
<tr>
<td>32</td>
<td>0.2641597092151642</td>
</tr>
</tbody>
</table>
Trapezoidal Rule

To integrate $f(x) = x^e$; $[0,1]$

Using $n$ trapeziums

<table>
<thead>
<tr>
<th>$N$</th>
<th>AREA</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>.243602529168129</td>
</tr>
<tr>
<td>4</td>
<td>.2590458346469879</td>
</tr>
<tr>
<td>8</td>
<td>.2629398107528687</td>
</tr>
<tr>
<td>16</td>
<td>.2639156579971313</td>
</tr>
<tr>
<td>32</td>
<td>.2641597092151642</td>
</tr>
<tr>
<td>64</td>
<td>.2642207741737366</td>
</tr>
<tr>
<td>128</td>
<td>.2642359435558319</td>
</tr>
</tbody>
</table>

The actual value is .2642411291599274

Press SPACE BAR to continue
Trapezoidal Rule

Since \( \text{Error} = \text{Actual} - \text{Calculated} \)

And \( \text{Error} \) is proportional to \( h \) square

Therefore; if \( N \) increases by 2; then \( \text{Error} \) decreases by 4

<table>
<thead>
<tr>
<th>N</th>
<th>Error</th>
<th>Error of ( n ) / Error of 2n</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2.06385999917984E-002</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5.196094512939453E-003</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1.301318407058716E-003</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>3.25471627960205E-004</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>8.141994476318359E-005</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>2.03549861907959E-005</td>
<td></td>
</tr>
</tbody>
</table>

Press SPACE BAR to continue
**Trapezoidal Rule**

Since \[ \text{Error} = \text{Actual} - \text{Calculated} \]

And \[ \text{Error is proportional to } h^2 \text{ square} \]

Therefore; \[ \text{if } N \text{ increases by 2, then Error decreases by 4} \]

<table>
<thead>
<tr>
<th>(N)</th>
<th>(\text{Error})</th>
<th>(\text{Error of } n / \text{Error of } 2n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>(2.06385999917984E-002)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>(5.196094512939453E-003)</td>
<td>(3.971944686610994)</td>
</tr>
<tr>
<td>8</td>
<td>(1.301318407058716E-003)</td>
<td>(3.99294629566014)</td>
</tr>
<tr>
<td>16</td>
<td>(3.254711627960205E-004)</td>
<td>(3.998260232579434)</td>
</tr>
<tr>
<td>32</td>
<td>(8.141994476318359E-005)</td>
<td>(3.997437774524158)</td>
</tr>
<tr>
<td>64</td>
<td>(2.03549861907959E-005)</td>
<td>(4)</td>
</tr>
</tbody>
</table>

Press SPACE BAR to continue
**Trapezoidal Rule**

As we can see, the values are very near 4 in the last column. However, rounding error begins to dominate as N gets large.

Try it yourself. Please input the starting N (1-9, 0 to quit)?

<table>
<thead>
<tr>
<th>N</th>
<th>Error</th>
<th>Error of n / Error of 2n</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>1.028329133987427E-003</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2.571940422058105E-004</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>6.431341171264648E-005</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>1.603364944580008E-005</td>
<td></td>
</tr>
<tr>
<td>144</td>
<td>4.02331352238867E-006</td>
<td></td>
</tr>
<tr>
<td>288</td>
<td>1.072883605957031E-006</td>
<td></td>
</tr>
</tbody>
</table>
Trapezoidal Rule

As a Summary:

1. Formula for Trapezoidal rule is simple
2. It needs n+1 function evaluations
3. Error is proportional to $h^2$ square for small N
4. Rounding error dominates as N becomes large.

Let us now study the Simpson's Rule.

Press SPACE BAR to continue
Simpson's Rule
Simpson’s Rule

The formula for Simpson’s Rule is given as:

\[
\text{Area} = h \left\{ f(a) + f(b) + 4 \left[ f(x_1) + f(x_3) + \ldots \right] \\
+ 2 \left[ f(x_2) + f(x_4) + \ldots \right] \right\} / 3
\]

And the truncation error is given as:

\[
\text{Error} = \frac{4}{3} h^4 M / 180
\]

Let's again use this formula to find the numerical integration of the same function and investigate the error terms:

\[
f(x) = xe^{-x} ; \quad \text{for } x \text{ between } 0 \text{ and } 1
\]

Press SPACE BAR to continue
Simpson's Rule

To integrate \( f(x) = xe^{-x} \); \([0,1]\)

Press SPACE BAR to continue
To integrate $f(x) = xe^{-x}$; $[0,1]$

Using even $N$ intervals

<table>
<thead>
<tr>
<th>$N$</th>
<th>AREA</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.2634901404380798</td>
</tr>
<tr>
<td>4</td>
<td>0.2641925513744354</td>
</tr>
<tr>
<td>8</td>
<td>0.2642380595207214</td>
</tr>
</tbody>
</table>
To integrate \( f(x) = xe^{-x} \); \([0,1]\)

Using even \( N \) intervals

<table>
<thead>
<tr>
<th>( N )</th>
<th>AREA</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.2634901404380798</td>
</tr>
<tr>
<td>4</td>
<td>0.2641925513744354</td>
</tr>
<tr>
<td>8</td>
<td>0.2642380595297214</td>
</tr>
<tr>
<td>16</td>
<td>0.264240950345993</td>
</tr>
<tr>
<td>32</td>
<td>0.264241099357605</td>
</tr>
<tr>
<td>64</td>
<td>0.264241099357605</td>
</tr>
</tbody>
</table>

The actual value is 0.2642411291599214

Press SPACE BAR to continue
**Simpson’s Rule**

Since \[ \text{Error} = \text{Actual} - \text{Calculated} \]
And \[ \text{Error is proportional to fourth power of } h \]
Therefore; if \( N \) increases by 2; then Error decreases by 16

<table>
<thead>
<tr>
<th>( N )</th>
<th>Error</th>
<th>Error of ( n ) / Error of ( 2n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>7.509887218475342E-004</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4.85778549194336E-005</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>3.069639205932617E-006</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1.788139343261719E-007</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>2.90232238769531E-008</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>2.90232238769531E-008</td>
<td></td>
</tr>
</tbody>
</table>
**Simpson's Rule**

Since \( \text{Error} = \text{Actual} - \text{Calculated} \)

And \( \text{Error} \) is proportional to fourth power of \( h \)

Therefore; if \( N \) increases by 2; then Error decreases by 16

<table>
<thead>
<tr>
<th>( N )</th>
<th>Error</th>
<th>( \frac{\text{Error of } n}{\text{Error of } 2n} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>7.509887218475342E-004</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4.85778549194336E-005</td>
<td>15.45950920245399</td>
</tr>
<tr>
<td>8</td>
<td>3.069639205932617E-006</td>
<td>15.8252427184666</td>
</tr>
<tr>
<td>16</td>
<td>1.788139343261719E-007</td>
<td>17.16666666666667</td>
</tr>
<tr>
<td>32</td>
<td>2.980232238769531E-008</td>
<td>6</td>
</tr>
<tr>
<td>64</td>
<td>2.980232238769531E-008</td>
<td>1</td>
</tr>
</tbody>
</table>

Press SPACE BAR to continue
**Simpson's Rule**

For small $N$, the values are near 16 in the last column.

However, rounding error begins to dominate as $N$ increases.

Try it yourself. Please input the starting $N (2-8, 0$ to quit)?

<table>
<thead>
<tr>
<th>N</th>
<th>Error</th>
<th>Error of n / Error of 2n</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>$3.069639205932617E-006$</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>$1.788139343261719E-007$</td>
<td>$17.166666666666667$</td>
</tr>
<tr>
<td>32</td>
<td>$2.980232238769531E-008$</td>
<td>6</td>
</tr>
<tr>
<td>64</td>
<td>$2.980232238769531E-008$</td>
<td>1</td>
</tr>
<tr>
<td>128</td>
<td>$2.980232238769531E-008$</td>
<td>-1</td>
</tr>
<tr>
<td>256</td>
<td>$2.980232238769531E-008$</td>
<td>-1</td>
</tr>
</tbody>
</table>
Simpson's Rule

As a Summary:

1. Formula for Simpson's rule is not complicated
2. It also needs $n+1$ function evaluations
3. Error improves significantly
4. Error is proportional to fourth power of $h$
5. Rounding error dominates quickly.

Press SPACE BAR to continue
Conclusion
Conclusion

Trapezoidal Rule and Simpson’s Rule have the following similarities

1. They are based on the Newton-Cote method.
2. They all use equal space width $h$.
3. $f(a)$ and $f(b)$ are in the formulas.
4. The formulas need $N+1$ function evaluations.

Press SPACE BAR to continue
Conclusion

However, the two formulas have the following differences:

1. Trapezoidal Rule
   1. It approximates a curve by continuous straight lines.
   2. Error is only proportional to \( h^2 \) square.
   3. Hence in general, it converges to the solution slower.

Press SPACE BAR to continue
II. Simpson's Rule

1. It approximates a curve by continuous quadratic lines.
2. Error is proportional to fourth power of $h$.
3. Hence in general, it converges to the solution faster.
4. Rounding error dominates quickly as $N$ increases.
Conclusion

This concludes the Numerical Integration lesson.

Please take out the disk and turn off the computer.
Do not forget to answer the question paper.
Please answer the following questions according to the package you use.

Section A

1. What is the difference between numerical integration and area under a curve?

2. In the Trapezoidal Rule, what is the relation between the error and $N$, the number of trapeziums used?

3. In the Trapezoidal Rule, for what $N$ does the ratio $\frac{\text{Error of } N}{\text{Error of } 2N}$ begin to diverge? What is that ratio?
4. In Simpson's Rule, what is the relation between the error and $N$, the number of intervals used?

5. In Simpson's Rule, for what $N$ does the ratio $\frac{\text{Error of } N}{\text{Error of } 2N}$ begin to diverge? What is that ratio?

6. Why does the ratio not obey the error term for large $N$?

Section B

1. How long did you take to finish the package? _________ min.

2. Can you type? If you can, how fast?

3. What computer experience do you have?

4. Do you have a personal computer?

5. Have you taken any similar package like this one before? ________

If you do, what package? ________________________________
Section C

Please use the following scale to circle your answers.

6 - strongly agree, 1 - strongly disagree

1. The procedure for using the package is simple. 6 5 4 3 2 1
2. The length of this package is about right. 6 5 4 3 2 1
3. The screens are easy to read. 6 5 4 3 2 1
4. Interaction between user and computer is adequate. 6 5 4 3 2 1
5. The content of this package is well presented. 6 5 4 3 2 1
6. The examples are easy to follow. 6 5 4 3 2 1
7. Accompanying written lesson notes are helpful. 6 5 4 3 2 1
8. A CAL package like this in Numerical Analysis is appropriate. 6 5 4 3 2 1
9. If time allows, I would like to have more computer packages available. 6 5 4 3 2 1
10. I would like to borrow a package and use it at home or in the office. 6 5 4 3 2 1

11. Please describe briefly anything you particularly like about this package.

________________________________________________________________________
________________________________________________________________________

12. Please describe briefly anything you particularly dislike about this package.

________________________________________________________________________
________________________________________________________________________
________________________________________________________________________

13. If another package were to be used, what topic in Numerical Analysis do you think should be on?

________________________________________________________________________
Section D - Please tear off this page, answer the following question later and hand it in together with your assignment.

Describe, in about 150 words, what do you think the advantages and disadvantages about using a package like this one in other subjects of your course.