THAI HIGH SCHOOL COMPUTER LITERACY:
A CONTENT ANALYSIS

DISSERTATION

Presented to the Graduate Council of the
University of North Texas in Partial
Fulfillment of the Requirements

For the Degree of

DOCTOR OF PHILOSOPHY

By

Pornpum Chaipraparl, B.Sc., M.Ed.
Denton, Texas
December, 1989

This study examined the extent to which each computer literacy objective domain, each specific mode of instruction, and each type of question were treated in Thai high school computer literacy text materials. Two textbooks and their accompanying teachers' manuals were examined using three analytical schemes as frameworks for the examinations. The Minnesota Educational Computing Consortium (MECC) computer literacy objectives were used to classify the content in the text materials in order to determine the degree of emphasis on each computer literacy objective domain. The Hawaii State Department of Education (HSDE) instructional modes were used to classify the content in the text materials in order to determine the degree of emphasis on each mode of instruction. Bloom's taxonomy of education, cognitive domain, was used to classify the review questions and exercises in the text materials in order to determine the degree of emphasis on each cognitive level. Detailed findings are given as numerals, percentages, and decimal values. Perspectives are offered on the need for textbooks which reflect the values and feelings objectives. Conclusions were that (a) text materials focus most on the
programming/algorithms objectives and tend to exclude the values and feelings objectives; (b) text materials use only three modes of instruction, focusing first on the topic mode, second on the tutee mode, and last on the tool mode; (c) text material questions focus more on higher cognitive than on lower cognitive levels.
ACKNOWLEDGEMENT

This study was supported by a grant from the Institute for the Promotion of Teaching Science and Technology (IPST). The assistance is acknowledged with appreciation.
TABLE OF CONTENTS

LIST OF TABLES ................................................. iii

CHAPTER

I. INTRODUCTION .............................................. 1

Statement of the Problem
Purpose of the Study
Research Questions
Significance of the Study
Definitions of Terms
Delimitations
Basic Assumptions
Organization of the Study

II. REVIEW OF RELATED LITERATURE ....................... 9

The Concept of Computer Literacy
Computer Literacy in the School Curriculum
Content Topics for Computer Literacy
Computer Applications
Computer Programming
Computer Knowledge
Social Implications of Computers
Instructional Objectives for Computer Literacy
Instructional Modes
Topic Mode
Tutor Mode
Tool Mode
Tutee Mode
Computer Literacy Textbooks
Computer Literacy in the Context of Thai Education
Structure of the Thai Educational System
Computer Education at the Secondary School Level
The IPST Computer Literacy Curriculum
Impediments to the Implementation of the Curriculum
# LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Number and percentages of coding units from the textbooks examined</td>
<td>47</td>
</tr>
<tr>
<td>2. Intercoder reliability of the coding unit classifications</td>
<td>53</td>
</tr>
<tr>
<td>3. Percentages of coding units devoted to instructional objectives in each content category classified according to the MECC analytical scheme</td>
<td>57</td>
</tr>
<tr>
<td>4. Percentages of coding units devoted to instructional objectives in application domain classified according to the MECC analytical scheme</td>
<td>59</td>
</tr>
<tr>
<td>5. Percentages of coding units devoted to instructional objectives in hardware domain classified according to the MECC analytical scheme</td>
<td>61</td>
</tr>
<tr>
<td>6. Percentages of coding units devoted to instructional objectives in impact domain classified according to the MECC analytical scheme</td>
<td>62</td>
</tr>
<tr>
<td>7. Percentages of coding units devoted to instructional objectives in limitations domain classified according to the MECC analytical scheme</td>
<td>64</td>
</tr>
<tr>
<td>8. Percentages of coding units devoted to instructional objectives in programming/ algorithms domain classified according to the MECC analytical scheme</td>
<td>66</td>
</tr>
<tr>
<td>9. Percentages of coding units devoted to instructional objectives in software and information processing domain classified according to the MECC analytical scheme</td>
<td>67</td>
</tr>
<tr>
<td>10. Percentages of coding units devoted to instructional objectives in usage domain classified according to the MECC analytical scheme</td>
<td>70</td>
</tr>
</tbody>
</table>
11. Percentages of coding units devoted to each instructional mode classified according to the HSDE analytical scheme. . . 72

12. Percentages of coding units devoted to topic mode by content category classified according to the HSDE analytical scheme. . . 73

13. Percentages of coding units devoted to tutee mode by content category classified according to the HSDE analytical scheme. . . 74

14. Percentages of coding units devoted to tool mode by content category classified according to the HSDE analytical scheme. . . 75

15. Percentages of coding units devoted to each cognitive level classified according to Bloom's cognitive level . . . . . . . . 77

16. Percentages of coding units devoted to knowledge level classified according to Bloom's cognitive level . . . . . . . . 78

17. Percentages of coding units devoted to comprehension level classified according to Bloom's cognitive level . . . . . . . . 79

18. Percentages of coding units devoted to application level classified according to Bloom's cognitive level . . . . . . . . 80

19. Percentages of coding units devoted to analysis level classified according to Bloom's cognitive level . . . . . . . . 81

20. Illustrative contingency table for agreement measure between two coders . . . . 120
CHAPTER I

INTRODUCTION

Because technology has always affected education and has determined the kind of knowledge an individual needs to participate effectively in society, the increased use of computer technology in contemporary society suggests the need to incorporate a computer literacy program into the educational system. If education is to prepare the next generation for effective participation in society, all students should be exposed to the nature and uses of computers so that they are prepared to cope in an increasingly technological environment. Students should be aware of the capabilities, limitations, and social importance of computer technology.

Thailand, like other countries, recognizes the need to produce computer-literate students. As a result, a computer literacy program was established and is included as part of the total school curriculum. The national computer literacy program was first implemented in high schools in 1985. Its major goal is to provide students with a basic understanding of computer technology and to provide simple programming ability. Since its implementation, computer literacy has become widely recognized among secondary schools. A number of schools, where microcomputers are available, have
included computer literacy programs in their curricula. However, the schools have faced several problems, such as inadequate microcomputer facilities for students, a shortage of good educational software, and a lack of adequately-trained teachers.

In light of these problems, textbooks have become the key element in the educational process, since they are crucial in achieving and maintaining standards of quality (Chall, 1981; Strahan & Herihy, 1985). Textbooks determine the manner in which instruction should be carried out, as well as the instructional objectives to be achieved. The content presented in the textbook, the kind of questions asked, and the manner in which the students are encouraged to get involved with the materials can influence how well students are able to learn from the texts (Doran & Sheard, 1974).

The rapid growth of computer technology suggests that the newly implemented computer literacy curriculum needs to be reexamined in order to determine whether goals and objectives should be changed, the materials being used are appropriate, and whether students are learning what is expected. The analysis of computer literacy text materials in Thailand is mandatory, since it helps delineate the strengths and shortcomings of the curriculum. It provides information useful in identifying the established educational objectives, specific modes of presentation
(Tamir, 1985, p. 87), and content areas covered. This information, thus, can be directly applied to the revision of course materials and the improvement of teaching effectiveness. Moreover, it can serve as a basis for the development of preservice and inservice computer education training programs.

Statement of the Problem

This study was an analysis of a computer literacy program as reflected in the materials used in high schools in Thailand.

Purpose of the Study

The specific purpose of this study was to analyze a computer literacy program in order to explore the existing national computer literacy program being offered to Thai students. The study focused on instructional objectives, specific modes of instructions, and how questions were used in the text materials.

Research Questions

To carry out the purpose of the study, answers were sought to the following questions:

1. To what extent do text materials focus on each computer literacy objective domain?

2. To what extent do text materials focus on each specific mode of instruction?
3. To what extent do text material questions focus on higher cognitive levels?

Significance of the Study

In Thailand, where the computer literacy curriculum has been implemented for the first time, the lack of adequately-trained teachers, computer hardware, and software presents a major obstacle to effective implementation of the curriculum. In this situation, textbooks are crucial to the standard of education, since they are the only major source of information for both teachers and students.

Because of the dominance of textbooks as an important facet of education, it seems to be aware of their contents. This is especially true in the areas of learning, student evaluation and curriculum planning. There are several educationally significant justifications for analyzing content, as Stewart (1981) pointed out in his content analysis of science education:

From a curriculum standpoint, analysis of science content can serve as the basis for selecting cognitive intended learning outcomes [Johnson, 1967]. Cognitive skills--what students are expected to do with the knowledge learned ... can be more systematically defined after the knowledge elements to be learned have been identified. From the perspective of student evaluation, an analysis of the instructional content may prove useful for the development of test items or test situations. From a research standpoint, an important reason for analysing instructional content is to allow a comparison to be made between what students know and what they are to be or have been taught (p. 171-172).
Although many aspects of textbooks could be examined, the questions and instructional techniques used in textbooks are of special importance to the development of curriculum materials and to those who use them. It seems reasonable that the questions and instructional modes of presentation should be examined in terms of cognitive levels and computer uses.

This study was primarily concerned with an analysis of computer literacy text materials used in Thailand and focused on the coverage of contents, specific modes of presentation, and the use of textbook questions. The study is significant in that it:

1. describes the existing national computer literacy program;

2. provides information for revision of the computer literacy curriculum, particularly text materials, being used in schools;

3. provides information for improvement of computer literacy teaching effectiveness;

4. provides a basis for the development of computer literacy training programs for both preservice and inservice teachers; and

5. fills a research gap, since there are a number of studies concerning the analyses of other subject area text materials, but none in computer literacy.
Definitions of Terms

The following terms have restricted meaning and are defined for use in this study.

Computer literacy is defined as whatever understanding, skills, and attitudes one needs to function effectively within a given role that directly or indirectly involves computers (Anderson & Klassen, 1981, p. 131).

Computer literacy objectives are desired learning outcomes for computer literacy courses.

High school level is an educational level which contains grades 10, 11, and 12.

Higher cognitive questions are questions that require a high level of cognitive skills from comprehension, application, analysis, and synthesis to evaluation (Bloom, 1956).

Instructional modes include four methods of computer literacy instruction: the computer as a tutor, the computer as a tutee, the computer as a tool, and the computer as a topic (Hawaii State Department of Education, 1985).

Text materials include curriculum materials in printed form such as textbooks and teacher manuals.

Delimitations

One limitation which must be recognized in this study is the fact that only textbooks and their accompanying teachers' manuals were used in the analysis. Other curriculum materials, such as transparencies, video tapes,
programmed texts, or other supplementary materials were not included in the study due to time and resource constraints.

Basic Assumptions

It was assumed that the analytical schemes used as frameworks for categorizing the text contents in this study were standard and valid.

Organization of the Study

The first chapter provides a general idea of the role of computer technology in society and the need to incorporate the computer literacy program into the school curriculum, particularly in Thailand. It also establishes the significance of the study and set forth research questions to be answered. The general theoretical foundation of the research has been presented.

The next chapter concentrates on the literature review which is composed of three sections: the concept of computer literacy, computer literacy in the school curriculum, and the computer literacy in the context of Thai education. This chapter is intended to provide the conceptual framework in this study.

The third chapter focuses on the method of data analysis. It provides a description of the instruments applied in this study, techniques in the coding process, and the method of analysis.
The fourth chapter focuses on the three research questions. The results are interpreted and explained using tables. Discussions are also included of each topic.

The last chapter includes a summary of findings, the conclusions, the implications of the study, and recommendations for further research. Discussion focuses on implications of the results.
CHAPTER II

REVIEW OF RELATED LITERATURE

This chapter begins with a presentation of the general concepts of computer literacy which are adopted differently by schools. Additional sections include a review of literature related to computer literacy in the school curriculum and computer literacy in Thai education.

The Concept of Computer Literacy

From the mid-1970s, the term computer literacy has become widely used and variedly defined by many experts. There is considerable disagreement as to what is meant by computer literacy. At one end of the continuum, it is generally defined as basic awareness and understanding of computers and their uses (Ball & Charp, 1977; O'Donnell, 1982). The Conference Board of the Mathematical Sciences (1972) defines computer literacy as an understanding of computer capabilities, applications, and algorithms. In a similar way, Moursund (1976), defines computer literacy as "the non-technical and low-technical aspect of the capabilities and limitations of computers, and of the social, vocational, and educational implication of computers" (p. 55). His definition also includes social issues in its terminology.
At the other end, computer literacy is equated with skills in programming and the operation of computers (Ershov, 1981; Luehrmann, 1982). Between these two extremes, the term computer literacy is defined as the ability to function effectively with the computer and its applications. Rawitsch (1978) points out that a functional facility with hardware and software as problem solving tools is a must for computer literacy. Coburn (1982) also describes computer literacy in terms of four components: the ability to program computers, the ability to use preprogrammed application of computers, the understanding of the role of computers in society, and the ability to use computers to solve problems. Eisele (1983) identifies five areas that contribute to computer literacy: computing skills for real-life problem-solving, computing skills for vocational and home use, ethical attitudes toward producing computer services, ethical attitudes toward consumption of computer services, and a positive attitude toward the role of computers in society. Watt (1980) explains computer literacy as "that collection of skills, knowledge, understandings, values and relationships that allow a person to function comfortably as a productive citizen in a computer-oriented society" (p. 26). Anderson and Klassen (1981) define computer literacy as "whatever understanding, skills, and attitudes one needs to function effectively within a given role that directly or indirectly involves
Their definition, even stated in the broadest sense, emphasizes an understanding of computers that describes an individual as being capable of evaluating computer applications as well as performing the applications. According to their view, individuals should be taught more than how to simply operate or program a machine. People also need to know how computers can be productively used and the consequences of computerization (Anderson, Klassen, & Johnson, 1982, pp. 19-20). Brumbaugh (1985) defines computer literacy as "an ability to define, demonstrate, and/or discuss: how computers are used, how computers do their work, how computers are programmed, how to use computers, and how computers affect our society" (p. 234). Simonson, Maurer, Montag-Torardi, and Whitaker (1987) incorporate existing definitions of computer literacy as the basis for the development of an instrument for assessing a computer literate person, which results in the following definition: an understanding of computer characteristics, capabilities, and applications as well as an ability to implement this knowledge in the skillful, productive use of computer applications suitable to individual roles in society (p. 233). This definition, implies that the knowledge and skills of a computer-literate person include knowledge of computer attitudes, computer applications, computer systems, and computer programming.
Obviously, the term computer literacy lacks a generally-accepted definition. From one end of the continuum to the other, the term covers aspects of learning about computers and of learning with, from, and through computers. It involves consideration of the application of computers in educational settings and in society at large, as well as consideration of implications for education and society (Anderson et al., 1986).

Computer Literacy in the School Curriculum

School computer literacy programs are found in two distinctly different forms: as a continuing part of the educational process encompassing many grade levels, and as a discrete course of learning units at specific grade levels. Within these two forms there are two basic approaches, either integrating the computer literacy material into the existing curriculum or presenting the material as a distinct course, separate from existing subject matter. It is generally agreed that the ideal approach is a continuing, integrated one. Computer literacy, like general literacy, is best acquired gradually, with exposure from many vantage points (Fritz, 1985, p. 705; Norton, 1988, p. 8).

The diversity of expert opinions as to what constitutes computer literacy results in a variety of school computer literacy programs, depending on the concept adopted by the schools. Ideally, once the concept of computer literacy has been adopted, it is translated into educational goals. From
these goal statements, more specific curriculum objectives typically emerge. Themes and contents are decided, and instructional objectives are formulated and used by teachers to guide students in the teaching of computer literacy. Thus, what is actually taught to students is often dictated by how those responsible define computer literacy (Bear, 1987).

**Content Topics for Computer Literacy**

Despite the diversity in concepts of computer literacy, some degree of consensus exists. The areas commonly included as parts of school computer literacy programs are computer application, computer programming, general computer knowledge, and societal implications.

**Computer Applications**

Computer applications offer rich opportunities for learning concepts as well as development of students' intellect. The applications have several components, including microworlds, which are cybernetic environments within which elements may be combined according to given rules; educational games, including adventure games, which develop reading comprehension and problem-solving skills; microcomputer-based instrumentation systems which permit explorations of real-world phenomena; data bases which increase information available to students; tools, including graph-plotting routines, word processing, spread-sheet
programs, and general purpose problem-solvers; special purpose computer languages, such as LOGO, which permit creation of learning environments that foster development of the child's intellect; simulations, which create flexible universes within which students may experimentally discover properties of the real world; and discovery learning in mathematics, which provides a more active and self-directed learning environment within which the student can discover properties of mathematical functions (National Science Board Commission on Precollege Education in Mathematics, Science and Technology, 1983, p. 53). The most important criterion for the successful introduction to computer applications is the amount of time a student actually spends working with computer applications to solve problems (California State Department of Education, 1985, p. 7). Computer applications are usually more effective with certain content areas such as language arts, particularly in the writing process; science; and mathematics (Roblyer, Castine, & King, 1988).

**Computer Programming**

Programming, or telling the computer what to do, has a role in the curriculum. Teachers and students need to have at least some knowledge of programming. One of the best arguments for teaching students computer programming is not to prepare students for computer-related careers, since the skills that they learn are likely to be obsolete by the time they graduate, but rather to assist in the development of
social and intellectual functioning. Some understanding of
how programs work gives students a more realistic
understanding of what programs can and cannot do and helps
them to function effectively in a society that is
increasingly dependent on information technology (Adams,
1985, pp. 94-95). This enables them to understand and use a
computer with existing software and to write programs for
explorations or use as tools to produce a product
(Sweetland, 1987, p. 3). Another argument of those equating
computer literacy with programming is that programming
fosters problem-solving and good logical thinking skills
(Blume, 1984; Luerhmann, 1984a; Mandell, 1982; Mayer, 1979;
Papert, 1980; Taylor, 1981; Watt, 1982). This argument is
supported by many studies (Bamberger, 1984; Fickel, 1986;
Horton & Ryba, 1986; Krasnor & Mitterer, 1984; Lewinta,
1986; Mann, 1986; Perkins & Bass, 1984; Rieber, 1987; Shaw,

A new approach to teaching programming places emphasis
on the development of the thinking process and
problem-solving ability, rather than on the syntax of
specific language and its rules. This can be achieved by
teaching structured programming languages (Lockard, 1986).
BASIC, LOGO, and PASCAL are the three main languages
currently taught in schools. BASIC is the most popular
programming language of teachers (Becker, 1983). BASIC is
the language that is built in all microcomputers, and is the
easiest language for beginners. Its shortcoming is that it does not encourage good programming techniques since it is not a structured language. Many who advocate teaching BASIC argue that, despite the lack of structure, BASIC can be effectively taught by using a structured programming approach (Anderson, Bennett & Walling, 1987; Lockard, 1986; Luehrmann 1984b; Luehrmann 1984c; Luehrmann 1984d; Luehrmann 1984e; Luerhmann, 1984f). LOGO is a procedural language that introduces students to programming with graphics but lacks necessary features for advanced programming. PASCAL is an advanced programming language that was developed in order to teach programming. It has all of the structures that are missing in LOGO or weak in BASIC (Sweetland, 1987, p. 4), but is somewhat difficult for beginning programmers (Tinker, 1983). LOGO and PASCAL are similar in that they are both procedural languages. Therefore, experiences from LOGO programming can be transmitted to PASCAL programming. As Sweetland (1987) suggests, those planning a computer curriculum should provide LOGO for beginning students and then allow students to move to PASCAL when they are ready to learn structures not available or easily understood with LOGO.

The major thrust of teaching students about computer programming is not only to present factual information, but also to provide a conceptual framework that can be used for reasoning and problem solving. The concrete conceptual
framework can be created through good programming environments (Gilbert-Macmillan, 1986) which include programming tools that help streamline exploration and discovery by students, such as trace debugers, error trapping routines, subroutine libraries, and file management routines. A survey of research on cognitive barriers to initial programming instruction by Wresch (1987) reveals that programming environments are critical to students' learning. He suggests that programming environments should be considered by instructors for the choices of instructional language (p. 163).

Computer Knowledge

A general knowledge of computers usually encompasses the appropriate, knowledgeable use of both hardware and software necessary for computer applications. More specifically, it includes some hands-on experience with computers, basic terminology relevant to computer system components, software, information processing, and data. Students should have at least a basic understanding of how computers work. Knowledge about hardware, software, and basic terminology, including some hands-on experience with computers, can help alleviate the mystique that exists about computers. Removing the mystery of computers can also remove some of the exaggerated value attached to anything that is done by computer (Fritz, 1985, p. 76).
Societal Implications of Computers

The present widespread use of computers necessitates an understanding by students of the impact of computer technology on their daily lives and on ethics and values related to computer uses. Important topics on the impact of computer technology include the historical development of computers, specific uses of computers in computer-related occupations and careers, limitations and social effects on proper and improper use of computers. Ethics and values concern issues of computer crime, copyrights, security, computer equity, job displacement, classroom computer rules, computerphobia, freedom of information versus privacy, artificial intelligence, etc. (Bear, 1986, p. 16).

Not much information is available regarding the extent to which this component of computer literacy is actually taught, when it is taught, and whether or not teaching is done effectively. The ACM Elementary and Secondary Schools Subcommittee (cited in East, 1981) makes the following suggestions in order to teach students about social and ethical concerns about computer usage in society:

To demonstrate, possibly by examples, what constitutes ethical and unethical conduct for computer professionals.

To define what is meant by "misuse" of the computer.

To provide possible resources for professionals and consumers who feel they have no place to turn with their computer-related ethics questions.

To establish the responsibilities of computer professionals in relation to those they interact with, e.g., employer, client, fellow professionals, members of the public (or society in general), and themselves.
To determine the rights and responsibilities of members of society related to computers and uses of computers (p. 58).

According to Aiken et al. (1982), students should learn about computing ethics as soon as they begin to actually use the equipment. In a situation where computer awareness is the primary goal, students should learn ethics after they learn the capabilities of the computer (p. 123). These authors add that teaching ethics can be effective when teachers demonstrate good computing ethics by serving as good examples and providing an appropriate atmosphere (p. 124).

Bear (1986) expresses his opinion about teaching computer ethics in a similar way:

It should be integrated throughout the unit, as well as throughout the regular classroom. Ethical and social implications should be discussed when the related issues are taught. For example, copyright issues should be discussed the first time a legal copy of a diskette is made; privacy, when databases are developed; artificial intelligence, when the history and the future of computers are explored. Class rules for computer use should be reviewed and discussed when students are first introduced to the computer in the classroom (p. 117).

In general, topics concerning the societal implications of computers have greater impact and relevance for students when dealt with in hands-on experiences rather than in isolation (California State Department of Education, 1985, p. 19).
Instructional Objectives for Computer Literacy

The establishment of educational objectives is essential to teaching and learning. If objectives are clearly defined, they can serve as a framework to help guide the processes of instruction and evaluation (Bloom, 1956, p. 21). A number of attempts have been made to identify instructional objectives for computer literacy in terms of cognitive, affective, and psychomotor behaviors. Neill (1977) identifies four major categories of instructional objectives for high school students. These include computer hardware, software, applications, and implications for society and individuals. He found that the highest rated objectives were those associated with topics related to the implications of computers for society and individuals, and the lowest rated objectives were those related to skills in designing and constructing computer hardware and computer systems.

The Department of Defense Dependents Schools (DoDDs) define instructional objectives in the areas of interacting with computers, functions and uses of computers, general problem-solving skills, and applications and impact in society in the computer literacy component in their computer curriculum (Grabriel, 1985). The California State Department of Education (1985) offers a perspective on the effective and creative use of computers and computer software throughout the school curriculum. The basic core
of skills covers the operation of computer systems; computer assisted learning; computer applications, including word processing, electronic spreadsheets, database managers, and programming; problem solving with computers, including communication skills and the thinking skills of describing the problem, breaking down the problem, developing algorithms, anticipating outcomes and finding unforseen results, generalizing a problem solution, and testing and debugging; computer science; and the societal impact of computers.

Ratekin et al. (1985) designed an instructional unit on secondary computer literacy courses covering the role of computers in society; ways in which computers affect society (historical development of computers, their effects on quality of life, economic effects of computers, concerns arising from computer use, and rapid technological changes in business); the way in which computers work (system components, binary code, kinds of computers, and differences between hardware and software); procedures for using computers (operation of a computer system, techniques in analyzing and solving a problem through flowcharting, procedures for writing and executing simple programs coded in BASIC, introduction to other languages, and software applications); and career opportunities in areas using computer technology.
An appropriate classification of the Minnesota Educational Computing Consortium (MECC) instructional objectives for computer literacy is suggested by Anderson and Klassen (1981). The objectives cover the domains of applications, hardware, impact, limitations, programming/algorithms, software and data processing, usage, values and feelings, expanding educational objectives: cognitive (Bloom, 1956), affective (Krathwohl, 1964), and psychomotor (Harrow, 1972). One interesting point of this classification is that exposure to computers, the usage domain, is separately identified as one of the computer literacy domains. As Anderson, Klassen, Krohn, and Smith-Cunnien (1982) found in an empirical study on one of the MECC computer literacy projects sponsored by the National Science Foundation: "It seems obvious that direct computer use and involvement in computer courses does lead to the acquisition of basic knowledge and understanding of computer concepts" (p. 28). Actually, instructional objectives for any computer literacy program are made up of a set of cognitive, affective, and psychomotor behaviors. Cognitive components of computer literacy range from knowledge level skills and behaviors to higher level understanding and applications (Randhawa & Hunt, 1984, p. 10). Attitudes, values, and motivation are mediating components of, and supplements to, cognitive skills and abilities (p. 2). Psychomotor components involve motor
skills for sequencing and execution of tasks on computers or computer terminals (Anderson & Klassen, 1981, p. 134).

**Instructional Modes**

There are two possible approaches to the teaching of computer literacy. One approach is general lecture or discussion of topics related to computers. The second approach includes three general categories for the use of computers which are described by Taylor (1980) as tutor, tool, and tutee. Based on the uses of computers, instructional computing in computer literacy can be divided into four modes: topic, tutor, tutee, and tool (Hawaii State Department of Education, 1985).

**Topic Mode**

In this mode, computer literacy is introduced at the beginning stage. It deals with instruction in the mechanics of computers—how to use them, how they function, and how they impact areas of science, technology and education as well as daily life. More specifically, it encompasses history, ethics, careers, elementary machine operations, and other topics normally included in instruction on computers (Knezek, 1988, p. 16). The computer is not directly used in this mode. Usually other audio visual aids such as transparencies, films, videotapes, and programmed texts are employed.
Tutor Mode

In the tutor mode, the computer is looked upon as an aid to, or surrogate for, the teacher. It functions as a "tutor" in a subject. It includes methods of instruction which use the computer to guide a student through a learning experience. This mode has often been called Computer Assisted Instruction (CAI), and usually includes drill and practice, tutorial, games and simulation, and problem-solving (Poirot, 1980; Steinberg, 1984). The term CAI is commonly used to describe what takes place in the tutor mode. At the simplest level, the computer presents some subject material, the student responds, the computer evaluates the response, and from the result of the evaluation determines what to present next. At its best, the computer tutor can maintain student records, present the subject matter with a wide range of detail, and, in an extensive and flexible manner, test the student on the material. Many programs and systems have been developed along traditional lines. Experts in programming and in the subject to be presented must prepare programs for the computers which have built-in solutions to the problems. Because they have very limited powers of reasoning most of the programs would not be able to solve these problems themselves (Bonnet, 1985, p. 188). Artificial intelligence techniques have been applied to CAI in order to make the teaching programs capable of solving the problems that are
given to the students at an expert level. Such intelligent programs include intelligent tutoring systems (Sleeman, 1982) and expert consulting programs (Thorkildsen, Lubke, Myette, & Parry, 1986).

**Tool Mode**

Using the computer as a tool can be thought of as an extension of the CAI approach. The most promising use of computers is their use as tools of instruction and as an environment within which learning can occur. In the tool mode, the computer serves as a utilitarian function in getting a job done. The particular program that is working within the computer determines the nature of the tasks that can be accomplished. This involves using computer tools in applications such as word processing, manipulation of data, information in databases, retrieval of information, electronic spreadsheets, which arise in existing subject areas (Fritz, 1985). Reasons for using computers as tools, other than to expose children to the tools of the information age, are to support concept learning and to provide opportunities for a wide range of critical thinking and problem-solving skills (Hunter, 1988). As a result, computer applications also improve students attitudes and increase their motivation to learn (Hawisher, 1986; Menis, 1984). In this application, students use computer programs to perform various tasks which would otherwise be too tedious or time-consuming. For example, students might use
the computer as a calculator in mathematics assignments, as a statistical aid in science experiments, or as a text editor in English. In the tool mode, the computer provides a service but does not function primarily as a teacher or tutor. Use of the computer in the tool mode may teach the user something, but the teaching is usually accidental and not the result of a design for teaching.

**Tutte Mode**

In the third mode, the computer becomes an object to be studied. The computer functions as a tutee, and the student as a tutor. The student or teacher doing the tutoring must learn to program or communicate with the computer. With this ability, the student teaches the computer to perform certain tasks, such as how to tutor or how to be a tool. For example, computers might be taught to tutor students in arithmetic, to calculate loan interest, or to generate animated graphics. Learners gain new insights into their own thinking through learning to program, and teachers enrich their understanding of education and broaden their view of how students can benefit from treating the computer as a tutee. As a result, extended use of the computer as a tutee can shift the focus of education in the classroom from end product to process and from acquiring facts to manipulating and understanding them.
Computer Literacy Textbooks

Despite many technologies which have been applied to classroom settings, textbooks remain the basic instructional materials in all subject areas. The role of textbooks cannot be over-emphasized. Their primary function is to inform students in the content areas (Anderson & Armbruster, 1984; Yeager & Edward, 1980). Most teachers consider the use of textbooks an integral part of the instruction they provide to students (Elliot & Clyde, 1980; Harms & Yagers, 1981; Metcalf, 1980; Ulerich, 1983) regardless of teaching techniques they use (Doran & Sheard, 1974; Yager, 1983; Yager & Stodgill, 1979). Because most students have a tendency to learn what is in the text and not learn topics not covered in the book (Begle, 1973), textbooks have an important influence on student learning. Spiegel and Wright (1984) concluded, in their study on preferences of textbook characteristics of biology teachers, that the content in the textbook, if accurate, up-to-date, and presented in a lively manner, promotes learning.

Besides presenting information, textbooks, through the use of questions, can give students practices in application of what they have learned and opportunities to bring together concepts from different fields (Davis & Hunkins, 1966; Trachenberg, 1974). The importance of textbook questions is evidenced by their presence in books, the suggestions for their use in the teachers' manuals, and by
their frequent use in the classroom. Textbook questions are used most frequently for introduction, motivation, and review and test exercises (Davis, & Hunkins, 1966). In general, questions placed within text materials appear to produce significantly higher performance than reading materials without questions. Questions placed after the reading materials have been found to be significantly more productive than prequestions, or questions placed immediately before the reading passages (Leonard & Lowery, 1984). In addition, higher level cognitive questions have been found to produce more learning than recall and factual questions (Rickards, 1974, 1976; Watts & Anderson, 1971). Indeed, questions in any textbook should be designed to accomplish objectives other than knowledge of specific facts in order to foster the development of students' thinking. The extent to which textbook questions emphasize higher intellectual processes could provide evidence concerning the cognitive objectives and their relative importance as indicated by the textbook.

Systematic descriptive studies of computer literacy text materials are rarely found in the literature. Nonetheless, suggestions for what to look for in computer text materials are found in a recent review of Texas' adopted computer literacy text materials for grades six through eight, conducted by Boudrot and Switz (1986). They offer guidelines which focus on teacher expertise, existing
curriculum, software, hardware, and student population, to aid in the evaluation of computer literacy textbooks in the following questions:

Teacher expertise—Will the majority of teachers have previous knowledge in computer literacy instruction? If the answer is no, complete lesson plans and software support may be necessary to keep the program afloat during the first few years. If the answer is yes, some sacrifices may be made with regard to lesson plans and support materials. More attention may be paid to the student text and its appeal to the general student population.

Existing curriculum—If a computer literacy curriculum already exists, is the structure flexible enough to modify to a textbook or must the textbook allow for various sequences? Also are there specific weaknesses in the curriculum that need support from the text?

Software—If the district is just beginning computer literacy instruction, accompanying software by several textbook publishers might solve the software selection problem. Districts should ask publishers' representatives if teacher training is available and, if so, at what cost. If the district has been using applications software, does the text support the existing software? Or is there sufficient investment in applications software that purchasing textbook-correlated software is not feasible? Site licensing and networkability may be important in light of tightened budgets.

Hardware—What brands of computer does each text support? Are there sufficient computers so each student will receive extensive computer experience? If the course is designed more toward awareness, a series that tends toward "off-line" activities would be more desirable.

Student population—Is the general student population experienced with computers? A more sophisticated book will be necessary for advanced users; a non-threatening book will serve a novice audience (p. 24).

Their questions have a great deal of intuitive appeal because they direct the evaluator's attention to aspects of computer literacy textbooks which provide the most satisfactory instructional base suitable to their needs.
They also point out the strengths and weaknesses of text materials based on topics covering all essential elements, which include computer-related terminology and use, history and development of computers, use of the computers as a tool, communicating instructions for the computer, and problems and issues of computer use in society, as required by the State of Texas. Among the strengths found in the text materials are:

1. Enough flexibility to allow for emphasis on applications, literacy on programming and accommodation of the availability or absence of computers.

2. The use of questioning strategies in the textbook to tap higher levels of thought.

3. Introduction of terminology in contexts that relates to real situations, not just as theoretical or technical jargon.


5. Provision of a well-balanced amount of information material.

6. The highlighting of main concepts by using appealing, colorful illustrations and diagrams.

7. The establishment of effective learning patterns for students by alternating teacher-directed lessons and hands-on activities.
8. Clear narrative lesson plans, and specific teacher-directed activities. The weaknesses include:

1. A lack of pictorial supports or illustrations to assist students with unfamiliar concepts.

2. A lack of depth in which new concepts are presented which may not be challenging enough for advanced students.

3. The use of questions which stress lower comprehension abilities that would discourage more advanced students.

4. Failure of objectives in the student book to match the objectives in the teacher's guide.

Computer Literacy in the Context of Thai Education

Structure of the Thai Educational System

The national scheme of education provides for six years of primary education, three years of lower-secondary and three years of upper-secondary education. Primary education is designed to provide and maintain literacy and to develop individual cognitive abilities, numeral manipulation, communication skills and adequate knowledge for future occupational roles. It also aims toward personal development and the promotion of attitudes desirable for life in a democratic society. It is compulsory for all Thai children. Secondary education aims to provide appropriate academic and vocational knowledge, compatible with the
learners' age, needs, interests, skills and aptitudes, which will ultimately be beneficial in a chosen career and for society. Extensive elective subjects in the academic and vocational areas are offered in the lower-secondary level. In the upper-secondary level, students are encouraged to concentrate on areas of specialization needed for their chosen career or occupation. Higher education aims at the full development of human intellectual abilities to facilitate advancement in knowledge and technology (United Nations Educational, Scientific, and Cultural Organization [UNESCO], 1984a, p. 361).

The educational system in Thailand is centralized under the supervision and control of the Ministry of Education and the Ministry of University Affairs. The primary, secondary and some higher education and non-formal education institutions are under the Ministry of Education, but most higher education institutions come under the Ministry of University Affairs (UNESCO, 1984b, p. 57).

**Computer Education at the Secondary School Level**

Computer education in Thailand first started at the university level around the mid-1960s. The advent of low-cost microcomputers in the late 1970s caused the use of computers to advance rapidly, which inevitably produced more and more pressure on the educational system to prepare students to cope with the technological environment. In response to the demand for computer-literate students, the
Ministry of Education established a policy for introducing computer education into the schools in 1983. Priority was given to the senior secondary years, due to limited resources and funds. The policy on the uses of computers in schools emphasized utilizing computers in teaching basic computer courses, assisting teachers in teaching various subjects using computer-assisted instruction techniques, and managing teachers' administrative routines (Ministry of Education, 1986). The Institute for the Promotion of Teaching Science and Technology (IPST) was given the assignment by the Ministry of Education to develop computer literacy courses for students in upper-secondary and vocational schools. The aim was to provide students with a basic understanding of computer systems and the ability to write a simple program (UNESCO, 1984b, p. 58).

The newly developed curriculum was implemented throughout the country in mid-1985. Two consecutive computer courses are currently offered to students as elective courses. Each course is studied for three periods per week per semester. The first course covers the area of computer awareness. The second course includes computer programming. For effectiveness of computer teaching, schools are required to procure central processing units, keyboards and monitors; at least one set for every five students, with no less than ten students enrolled in each computer course (UNESCO, 1984b, p. 58-60).
The IPST Computer Literacy Curriculum

The computer curriculum was developed by using the Institute for the Promotion of Teaching Science and Technology curriculum development model. Major activities include defining goals and objectives; deciding themes, content, skills and attitudes to be developed; developing curriculum materials; conducting school trials of materials; collecting data from trials; revising materials; training teachers; implementation; follow-up; and evaluation (UNESCO, 1984a, p.366).

The computer curriculum encompasses an awareness component in which students are expected to develop an understanding of the concepts of computers, and a programming component in which students are expected to develop logical thinking and problem-solving skills. The specific objectives of the curriculum are to develop (a) a basic understanding of a computer system; (b) perceptions of appropriate work for computers; (c) the ability to apply problem-solving algorithms to the work by computer; (d) the knowledge of programming and the ability to write simple programs; (e) the ability to run a simple computer program; and (f) logical thinking, systematic thinking, carefulness and creativity (UNESCO, 1984b, p. 59-60).

Contents included in the computer curriculum are based on foreign texts for universal topics. The first course which encompasses computer awareness covers the areas of
computer history, the computer's impact on society, applications of computers, knowledges of various types of computers and computer languages, computer vocabulary, and computer generations. The second course encompasses simple programming in the BASIC language.

Curriculum materials which are developed by IPST include students' books, teachers' manuals, and audio-visual aids such as videotape, slides, transparencies and programmed texts. The students' books contain examples of important concepts, hands-on activities with computers, and review questions and exercises. The teachers' manuals suggest teaching activities, illustrate how major concepts may be taught, and provide teachers with additional background materials and information about the course.

Trials of the two computer courses began in three pilot schools in 1984. All text materials were evaluated and revised. Prior to the implementation of each course, training workshops for a staff of trainers were conducted. Emphasis was placed on content and on the methodologies that teachers were expected to use in actual classroom settings. The IPST not only trained the staff of trainers but also collaborated with Science Teacher Servicing Centers (STSC), associated with teachers colleges throughout the country, in organizing inservice training programs. Instructional materials such as teachers' manuals and other related course materials are available directly from IPST.
School follow-up programs are continually conducted along with the curriculum implementation by means of various methods. Information regarding the implementation of the curriculum are collected through questionnaires. A team of IPST staff members visit schools to observe teachers to help with problems. Regular meetings are held to discuss strategies with follow-up personnel. Reports from direct observations of school visits and from questionnaires provide information for further revision and modification of all curricular materials.

Several activities are in operation in order to enhance the uses of computers in schools. Supplementary materials for existing courses include courseware for teaching science and mathematics which are being developed.

Efforts have been made to introduce computer literacy courses at the lower-secondary school level. There is also a plan to develop supplementary courses for vocational-oriented students.

Impediments to the Implementation of the Curriculum

The hardware problem--Only large schools or those supported by Parent and Teacher Associations or computer vendors can provide microcomputer facilities for students. Many schools cannot afford them and have not received financial support from the government. In schools where microcomputers are available they are often incompatible, they do not have exactly the same programming language, and
do not have the same standard operating system.

The software problem--Educational software is often unavailable or incompatible. Most available software is imported and cannot be easily adapted to Thai classroom activities because of language barriers and cultural bias. There is currently, no standard computer system that supports the Thai language. Efforts have been made to produce software written in Thai using graphic displays, but it is not suitable for educational purposes because it requires a lot of memory and has little editing flexibility.

The teacher training problem--There is a shortage of qualified trainers and adequately-trained teachers to teach new computer courses in schools, or to use microcomputers in teaching any subject. Teachers who are currently teaching did not learn about computers when they were learning how to be teachers. At present, only the IPST and the Department of Teacher Education are actively engaged in computer training activities.
CHAPTER III

METHOD OF ANALYSIS

This chapter describes the method used in this study and emphasizes on three issues: the coding procedure, the reliability of the coding procedure used in data analysis, and the statistical procedure. Systematic techniques were used in the coding process in order to obtain the most reliable data for further statistical analysis. Detailed descriptions of three analytical schemes are also provided. Statistical procedures are outlined along with the research questions.

Content analysis was used in this study because of its advantages over other techniques. As defined by Berelson (1952), content analysis is a research technique for the objective, systematic, and quantitative description of the manifest content of communication (p. 18). By using content analysis, there is less chance that errors from data collection processes will contaminate the data being analyzed because communication, which is text materials in this study, are usually nonreactive (Borg & Gall, 1983, p. 521; Kaid & Wadsworth, 1989). Compared with other techniques, such as interviews, experiments, questionnaires, and projective tests which are susceptible to these errors (Krippendorf, 1980, p. 29), content analysis usually yields
unobtrusive measures in which neither the sender nor the receiver of the message is aware that it is being analyzed. There is little danger that the act of measurement itself will act as a force for change and confound of the data (Weber, 1985, p. 10).

Sample of the Content Studied

Only the set of computer literacy textbooks and their accompanying manuals developed by the Institute for the Promotion of Teaching Science and Technology (IPST), Ministry of Education, were used in this analysis since there are no other text materials available for use. These materials are:


Textbook B and Teacher Manual B refer to the student book entitled Introduction to BASIC Programming and its accompanying manual. The content topics and instructional objectives for each chapter of these materials are presented in the Appendix A.

Instruments

Three analytical schemes were employed in this study: the Minnesota Educational Computing Consortium (MECC) computer literacy instructional objectives, the Taxonomy of Educational Objectives: Cognitive Domain (Bloom, 1956), and the Hawaii State Department of Education (HSDE) instructional modes (1985).

The MECC Computer Literacy Objectives

The MECC computer literacy objectives used in this study were adopted from the Computer Literacy Instructional Modules (CLIM) Project at MECC, which is based on the premise that computer literacy consists of a broad understanding of the role of computers in society as well as the ability to communicate with computers (Anderson, 1982, p.211). These objectives are a revision of the earlier computer literacy objectives that were developed for assessing computer literacy and awareness of senior high school students (Klassen, Anderson, Hansen, & Johnson, 1980). The original set of objectives included only six dimensions of computer literacy. They were hardware,
software, programming and algorithms, applications, social impact, and affect. The affective or attitudinal dimensions were further subdivided into such elements as computer enjoyment, computer anxiety, and self-confidence in computing (Anderson & Klassen, 1981, p. 132). Two dimensions on limitations and usage, were added and were used in the development of computer literacy instructional modules of the CLIM project. Each module was designed to be added as a supplementary unit to the usual science, social studies, and mathematics courses in junior and senior high school. The entire set of these instructional modules was equivalent to a two-semester course at the junior high level.

The revised objectives for computer literacy are specified in terms of instructional objectives and are categorized in eight dimensions or domains of application, hardware, impact, limitations, programming/algorithm, software and data processing, usage, values and feelings. The objectives in each domain are numbered with two values: the first value designates the objectives as elementary (1) and advanced (2), and the second is a sequencing number. The detailed description of the revised objectives are presented in Appendix B (Anderson & Klassen, 1981, pp. 143-150). These objectives are summarized as:

Applications--This domain covers the multitude of social and organizational areas into which computers have
been integrated. It also covers the general considerations for applying computers to new situations.

Hardware--This domain deals with the basic vocabulary of computer system components.

Impact--This domain of computer literacy encompasses the social effects of computerization, including both the positive and negative impacts of computers on society.

Limitations--This domain focuses on developing a general sense of capabilities and limitations of computing machines. Examples of computer limitations include the fact that computers do not have feelings or consciousness, and are not able to make value judgements.

Programming/Algorithms--This domain deals with the ability to read, modify, and construct algorithms and programs.

Software and Information Processing--This domain includes vocabulary relevant to software, information processing, and data.

Usage--This domain involves motor skills for sequencing and execution of tasks on computers or computer terminals.

Values and Feelings--This affective domain centers on developing positive attitudes toward personal use of computers as well as balanced attitudes toward computers as a social force.

Since this study aims at identifying instructional objectives from the content coverage of the computer
literacy text materials for high school students, these
domains of instructions are adopted in the analysis to
typify instructional objectives as they are exhibited
toward a given body of content types. Identifying the
content itself, in whatever raw form it exists, may not be
considered classifiable without considering the intended
instructional objectives.

The HSDE Instructional Modes

The categories of instructional computing in computer
literacy defined by the Hawaii State Department of Education
(HSDE) (1985, pp. 11-12) were used to classify modes of
instructions. The categories include topic, tutor, tutee,
and tool, which are described as:

Topic--The computer as a topic deals with instruction
in the mechanics of computers, how computers function and
their impact on the areas of science, technology and
education as well as daily life. With this mode, the
computer is not directly used. The facets of study include
structure and functions of computer as machines, history of
computer, career opportunities in computers, social impact
of the computer, value of computer to self and society, and
uses of the computer in society.

Tutor--The computer as a tutor includes methods of
instruction which use a computer to guide a student to a
lesson. These include drill and practice, games and
simulation, testing and response, and tutorial instruction.
Tutee--The computer as a tutee includes method of instruction using computer programming techniques by student who instruct the computer to perform a specific task. These include problem solving, algorithm construction, flowcharting, and programming.

Tool--The computer as a tool includes methods of using the computer to provide a service to the student. It acts as an instructional aid. Many of these are teacher management uses from which the students receive direct output. In others, students use the computer as a simple processing device as they would a calculator or a type writer. These include test scoring, attendance, curriculum material generation, grade keeping, material management, communications, data processing, and word processing.

**Taxonomy of Educational Objectives:**

*Cognitive Domain*

The categories drawn from Bloom's Taxonomy that are classified in hierarchical order from knowledge, comprehension, application, analysis, synthesis, and evaluation were used to categorize the review questions and exercises in text materials (Bloom, 1956, pp. 201-208). Questions or exercises that fell into categories of comprehension, application, analysis, synthesis, and evaluation were considered higher cognitive questions. All of these categories can be briefly described as:
Knowledge--The knowledge level implies recall or recognition of specific elements in a subject area, and can be achieved by rote memory of simple rules, facts, terminologies, sequences, and principles.

Comprehension--The Comprehension involves three different operations: translation, interpolation, and extrapolation. Translation is the lowest order in which the known concept or message is reinstated from one kind of symbology to another. Interpolation involves the translation of elements with the additional requirement that the student see interrelationships among elements of the total message. Extrapolation calls for going beyond the limits of the literal message and involve both translation and interpolation.

Application--The application, as states by Bloom, is the uses of abstractions in particular and concrete situations. The abstractions may be in the form of general ideas, rules of procedures or generalized methods. The abstraction may also be technical principles, ideas, and theories which must be remembered and applied. In other words, application is the ability to apply principle or generalizations to new problems and situations.

Analysis--The analysis involves the breakdown of a communication into its constituent elements, finding assumptions, identifying causal relationships, and distinguishing facts from opinions. Such analyses are
intended to clarify the communication, to indicate how the communication is organized, and the way in which it manages to convey its effects, as well as its basis and arrangement.

Synthesis--The synthesis level is the putting together of elements and parts so as to form a whole. This involves the process of working with pieces, parts, elements, etc., and arranging and combining them in such a way as to constitute a pattern or structure not clearly there before.

Evaluation--The highest level in Bloom's cognitive domain which requires some competence in all previous categories: knowledge, comprehension, application, analysis, and synthesis. Evaluation is the making of judgements about the value, for some purposes, of ideas, works, situations, methods, material, etc. It involves the use of criteria as well as standards for appraising the extent to which particulars are accurate, effective, economical, or satisfying. The judgements may be either quantitative or qualitative, and the criteria may be either those determined by student or those which are given to him.

Procedure for Coding Data

The coding unit for this study is a sentence. A set of rules for designating the coding unit were constructed (see Appendix C) since the reliability of a content analysis study depends not only on reliable rules for categorizing units but also on reliable rules for designating these units (Auld, & White, 1956). All directive statements, review
questions or exercises, illustrations, miscellaneous sentences, or examples contained in each chapter of the students' books were examined and pre-unitized by marking units with "slashes" (/) in order to improve the coding reliability (Stiles, 1978). A total of 1,684 units from the two textbooks were coded. Coding units from Textbook A, Introduction to Computers, totaled 873 or 52%; and units from Textbook B, Introduction to BASIC Programming, numbered 811 or 48%. The detailed breakdown of numbers and percentages of all coding units classified as directive statement, illustration, example, review question or exercise, and miscellaneous sentence are shown in Table 1. These coding units served as reference points for

Table 1

Number and percentage of coding units from the textbooks examined

<table>
<thead>
<tr>
<th>Content Unit</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Directive statement</td>
<td>586</td>
<td>67.1</td>
<td>207</td>
</tr>
<tr>
<td>Illustration</td>
<td>71</td>
<td>8.1</td>
<td>37</td>
</tr>
<tr>
<td>Example</td>
<td>74</td>
<td>8.5</td>
<td>353</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>48</td>
<td>5.5</td>
<td>51</td>
</tr>
<tr>
<td>Question or exercise</td>
<td>94</td>
<td>10.8</td>
<td>163</td>
</tr>
<tr>
<td>Total</td>
<td>873</td>
<td>100.0</td>
<td>811</td>
</tr>
</tbody>
</table>
further categorizing by applying the three analytical schemes as follows:

1. By using the MECC instructional objectives for computer literacy, each of all coding units that could be identified as meeting the instructional objectives was categorized as application, hardware, impact, limitations, software and data processing, programming/algorithms, usage, and values and feelings. The coding unit that did not fall into any category or subcategory were assigned to "others" category or subcategory. Such coding units were analyzed along with those in particular categories or subcategories. Examples of coding units among those examined in this study that fall into these categories or subcategories are illustrated:

Application:

A.1.2 Computers are capable of controlling manufacturing process, increasing number of products, controlling the quality of product to meet standard requirement as well as providing information useful for manufacturing products. (p.8, chapter 1, Textbook A).

Hardware:

H.1.12 Examples of output devices are display screen commonly called a CRT or cathode ray tube, printer, magnetic tapes, etc. (p. 30, chapter 2, Textbook A).

Impact:

I.2.5 Computers have great impact on the advance of research in science and technology since computerized data processing made it possible to deal with highly complicated data which is very difficult to carry out--sending out a satellite or a spacecraft, and designing of a huge building may
not be ever accomplished without computers. (p. 124, chapter 7, Textbook A).

Limitations:

L.1.1 Explain and reason whether a computer can think by itself. (p. 128, chapter 7, Textbook A).

Programming/Algorithms:

P.2.1 Modify the following program in order to print the result of AB + AC:

10 LET A=6
20 LET B=A
30 LET C=7
40 END

(p. 32, chapter 4, Textbook B).

Software and Information Processing:

S.2.3 Collect the data of all students in your classroom that includes their names, heights, weights, ages, and birthdates, and identify appropriate data type of each. (p. 50, chapter 3, Textbook A).

Usage:

U.1.3 LIST is a system command that directs the computer to display the stored programmed lines on screen. (p. 9, chapter 2, Textbook B).

Values and Feelings:

(No coding units identified in this category)

2. By using the HSDE instructional modes, each of the coding units were reexamined in conjunction with activities suggested in the accompanying teachers' manuals. They were further categorized as topic, tutor, tutee, and tool.

Examples of coding units that fall into these categories are illustrated:

Topic:

The fourth generation of computers can process data
in a picosecond. (p. 19, chapter 1, Textbook A).

As suggested in the teacher manual, teachers describe generations of computers by lecture method. Video-tape is used as an instructional aid (p. 2-3, chapter 1, Teacher Manual A).

Tutor:

(No coding units identified in this category)

Tutee:

Example 2

10 FOR TURN=5 TO 10
20 PRINT TURN
30 NEXT TURN
40 END

when run the program, the computer will display the following result:

5
6
7
8
9
10

(p. 45, chapter 6, Textbook B).

As suggested in the teacher manual, both teacher and students try programming on the microcomputers (p. 2-4, chapter 6, Teacher Manual B).

Tool:

Data is stored in computer memory in small units called bit which represents a binary digit and byte or character which is usually a set of eight binary digits or bits. (p. 39, chapter 3, Textbook A).

As suggested in the teacher manual, a microcomputer is used as a tool to generate the Binary Program to enable students to understand the binary coding system (p. 4, chapter 3, Teacher Manual A).

3. By using Bloom's cognitive level, only the coding units that are questions were categorized as knowledge, comprehension, application, analysis, synthesis, and
evaluation. Initially, questions from each textbook, primarily found at the end of chapters, were categorized into eight content domains, and then the questions were reexamined and assigned to their proper cognitive levels. Examples of the coding units that fall into these categories are illustrated.

Knowledge:

Identify at least three major components of a computer system. (p. 34, chapter 2, Textbook A)

Comprehension:

Describe each step in developing the program to calculate the area and the circumference of a circle whose radius is 3.0 units.

(p. 93, chapter 5, Textbook A)

Application:

Use the BASIC statement to compute the result of $2^5 + 3^4$. (p. 24, chapter 3, Textbook B)

Analysis:

Write the program to read three students' names and display the names in alphabetical order. (p. 64, chapter 7, Textbook B)

Synthesis:

(No coding units identified in this category)

Evaluation:

(No coding units identified in this category)

All of these categorizing units were assigned digit or character codes for further analysis. Computations were made by utilizing the statistical package for Statistical Analysis System (SAS) available at the computing center,
Reliability of Coding Procedure

If content analysis is to satisfy the requirement of objectivity, measures and procedures must be reliable. The following procedure was used to assess the investigator's accuracy of judgement in classifying the coding units to their appropriate categories or subcategories. Since a subsample of 5 to 7 per cent of a large sample is sufficient for assessing reliability (Stemple, 1981), a sample of 84 coding units out of 1684 units was randomly selected and independently categorized by the second coder and the investigator. The investigator classified the remaining coding units alone. An extensive training session was provided to the second coder in order to facilitate the application of analytical schemes to the sampling unit classification. The investigator explained the study in general, and the coding instrument, in detail. In the session, definitions for each category or subcategory were presented, many illustrations and examples of the categories were provided, and the second coder learned the procedure for marking the coding sheet. Since reliability is usually measured by the proportion of agreement of all categories identically coded by different coders, intercoder agreement was established between the investigator and the second coder to assess the reliability of the coding process. Cohen's (1960) coefficient called Kappa was used as a
measure of intercoder reliability. The statistic Kappa ($K$) is defined as:

$$K = \frac{(P_o - P_c)}{(1 - P_c)}$$

where $P_o$ is the observed proportion of agreement, and $P_c$ is the proportion of agreement expected by chance. The information on independent classification by the investigator and the second coder on the same randomly selected coding units according to the three analytical schemes is displayed in Table 2.

Table 2

<table>
<thead>
<tr>
<th>Analytical Scheme</th>
<th>$K$</th>
<th>S.E.</th>
<th>$z$</th>
<th>$p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MECC Computer Literacy Objectives</td>
<td>0.89</td>
<td>0.0532</td>
<td>16.80</td>
<td>0.000</td>
</tr>
<tr>
<td>HSDE Instructional Modes</td>
<td>0.90</td>
<td>0.0993</td>
<td>9.05</td>
<td>0.000</td>
</tr>
<tr>
<td>Bloom's Cognitive Levels</td>
<td>0.83</td>
<td>0.1078</td>
<td>7.75</td>
<td>0.000</td>
</tr>
</tbody>
</table>

The kappa ($K$) statistic is 0.83 for the MECC scheme, 0.90 for the HSDE scheme, and 0.83 for Bloom's cognitive level scheme. According to Kaid and Wadsworth (1989), researchers can usually be satisfied with coefficients over +0.85, those with below +0.80 should be suspect (p. 209). Thus, the level of reliability obtained represents fairly high agreement between the coders. Detailed computation of
the reliability measure and its relevant statistics are illustrated in Appendix D.

Procedures for Analysis of Data

The data in this study were analyzed by counting the number of coding units devoted to each selected aspect of instructional objective domains, instructional modes, and cognitive levels of text material questions and exercises. In other words, frequency count in both absolute and relative forms were applied to all research questions as follows:

Research Question 1: The extent to which the textbooks focus on each domain of computer literacy objectives was assessed by means of the MECC computer literacy instructional objectives scheme. The total number of coding units devoted to each category or subcategory determine the degree of emphasis on each content domain. An in-depth analysis was conducted in order to describe how instructional objectives are presented in each content domain across the two textbooks.

Research Question 2: The extent to which the text materials focus on each specific mode of instruction was evaluated by using the HSDE instructional computing categories. The total number of the coding units devoted to each category of the instructional modes determine how the information is presented in the text materials.
Research Question 3: The extent to which the textbook questions focus on higher cognitive levels was determined by the use of the six levels of Bloom's taxonomy of educational objectives. The total number of coding units, which are questions primarily found at end of chapters that fall into each cognitive levels, determines how questions are used in the text materials.

After all computations were made, the data were tabulated for ease of reporting and interpretation. These tables are presented in the next chapter.
CHAPTER IV

DATA ANALYSIS

This chapter presents the data analysis and findings of the study. The first two parts deal with the extent to which the text materials focus on computer literacy instructional objectives and on modes of instruction. The last part deals with the extent the text material questions focus on higher cognitive levels. In each part, tables which illustrate the findings are provided along with the research question and an explanation of statistical results.

PART 1: To what extent do the text materials focus on each computer literacy objective domain?

Statistical results of the classification of text materials according to the MECC instructional objectives for computer literacy are shown in Table 3. Within the textbook, the number of coding units representing instructional objectives contained in Book A indicated, in descending order, a degree of emphasis on programming/algorithms, impact, software and information processing, application, hardware, limitations, and usage content domains. None of the coding units could be identified as meeting the instructional objectives in the values and feelings domain. Book B indicated the highest degree of emphasis on programming/algorithms and a very low degree of
Table 3

Percentages of coding units devoted to instructional objectives in each content category classified according to the MECC analytical scheme.

<table>
<thead>
<tr>
<th>Content Category</th>
<th>Book A</th>
<th></th>
<th>Book B</th>
<th></th>
<th>Composite</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
<td>%</td>
<td>f</td>
<td>%</td>
</tr>
<tr>
<td>Application</td>
<td>144</td>
<td>16.5</td>
<td>62</td>
<td>7.6</td>
<td>206</td>
<td>12.2</td>
</tr>
<tr>
<td>Hardware</td>
<td>132</td>
<td>15.1</td>
<td>8</td>
<td>1.0</td>
<td>140</td>
<td>8.3</td>
</tr>
<tr>
<td>Impact</td>
<td>169</td>
<td>19.4</td>
<td>-</td>
<td>-</td>
<td>169</td>
<td>10.0</td>
</tr>
<tr>
<td>Limitations</td>
<td>32</td>
<td>3.7</td>
<td>-</td>
<td>-</td>
<td>32</td>
<td>1.9</td>
</tr>
<tr>
<td>Programming/Algorithms</td>
<td>217</td>
<td>24.9</td>
<td>631</td>
<td>77.8</td>
<td>848</td>
<td>50.4</td>
</tr>
<tr>
<td>Software and Information</td>
<td>163</td>
<td>18.6</td>
<td>27</td>
<td>3.3</td>
<td>190</td>
<td>11.3</td>
</tr>
<tr>
<td>Processing</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Usage</td>
<td>16</td>
<td>1.8</td>
<td>83</td>
<td>10.23</td>
<td>99</td>
<td>5.9</td>
</tr>
<tr>
<td>Values and Feelings</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td>873</td>
<td>100.0</td>
<td>811</td>
<td>100.0</td>
<td>1684</td>
<td>100.0</td>
</tr>
</tbody>
</table>

emphasis on usage, application, software and data processing, and hardware. None of the instructional objectives were found in the impact, limitations, and values and feelings domains. Considering the total program as a whole, about 94.1% of the coding units were devoted to cognitive instructional objectives that include application, hardware, impact, limitations, programming/algorithms, and software and information processing; and 5.9% devoted to psychomotor objectives, usage. None were found in the affective instructional objectives, values and feelings. The result also shows that the distribution of the amount of
instructional objectives among these content categories is not in a balanced proportion. It places most emphasis on programming/algorithms (50.4%) in particular, and less emphases on the usage (5.9%), and limitations (1.9%).

Beginning with Table 4, several analyses follow which provide a look at the distribution of instructional objectives within each content category. The results of analysis in the application domain, which covers the multitude of social and organization areas into which computers have been integrated, including general considerations for applying computers to new situations are shown in Table 4. All coding units in Textbook A could be identified as meeting the MECC instructional objectives in all subcategories except subcategories A.1.4 (recognize the definition of and some advantages of computer simulation) and A.2.3 (design and develop a computer-supported application program) which were not presented in the textbook. Emphasis seems to be placed on subcategories A.2.4 (recognize some general criteria for computer usage for a particular task), A.1.2 (recognize specific uses of computers and computer-supported applications), and A.1.7 (recognize that innovations in computer hardware and software continually expand the potential utility of the computer). In Textbook B, approximately 92% of the coding units represent instructional objectives in subcategory A.2.3 which were not found in Book A, and 8% found in
Table 4

Percentages of coding units devoted to instructional objectives in application domain classified according to the MECC analytical scheme.

<table>
<thead>
<tr>
<th>Content Category</th>
<th>Sub-category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f  %</td>
<td>f  %</td>
<td>f  %</td>
<td>-----------</td>
</tr>
<tr>
<td>Application</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A.1.1</td>
<td>6  4.2</td>
<td>-</td>
<td>6  2.9</td>
<td></td>
</tr>
<tr>
<td>A.1.2</td>
<td>24 16.7</td>
<td>-</td>
<td>24 11.7</td>
<td></td>
</tr>
<tr>
<td>A.1.3</td>
<td>9  6.3</td>
<td>-</td>
<td>9  4.4</td>
<td></td>
</tr>
<tr>
<td>A.1.4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>A.1.5</td>
<td>9  6.3</td>
<td>1  1.6</td>
<td>10  4.9</td>
<td></td>
</tr>
<tr>
<td>A.1.6</td>
<td>2  1.4</td>
<td>1  0.6</td>
<td>2  1.0</td>
<td></td>
</tr>
<tr>
<td>A.1.7</td>
<td>11 7.6</td>
<td>4  5.6</td>
<td>15 7.3</td>
<td></td>
</tr>
<tr>
<td>A.1.8</td>
<td>4  2.8</td>
<td>-</td>
<td>4  1.9</td>
<td></td>
</tr>
<tr>
<td>A.1.9</td>
<td>2  1.4</td>
<td>-</td>
<td>2  1.0</td>
<td></td>
</tr>
<tr>
<td>A.1.10</td>
<td>2  1.4</td>
<td>-</td>
<td>2  1.0</td>
<td></td>
</tr>
<tr>
<td>A.2.1</td>
<td>16 11.1</td>
<td>-</td>
<td>16 7.8</td>
<td></td>
</tr>
<tr>
<td>A.2.2</td>
<td>8  5.6</td>
<td>-</td>
<td>8  3.9</td>
<td></td>
</tr>
<tr>
<td>A.2.3</td>
<td>-</td>
<td>57 91.9</td>
<td>57 27.7</td>
<td></td>
</tr>
<tr>
<td>A.2.4</td>
<td>51 35.4</td>
<td>-</td>
<td>51 24.8</td>
<td></td>
</tr>
</tbody>
</table>

| Total            | 144 100.0    | 62 100.0| 206 100.0|

subcategories A.1.7 and A.1.5. More than half of the 206 coding units from both books were in advanced levels, the rest were in elementary levels. The text materials reflect all but one of the instructional objectives in this content domain as dealing with the definition and advantage of computer simulation (A.1.4). This is due to the approach used to introduce computer literacy into the existing curriculum as separate courses, not as extracurricular
activities integrated into other subject areas, thus, there seems to be no place for simulations or other forms of CAI programs.

Illustrated in Table 5 is an analysis of instructional objectives in the hardware domain which deals with basic vocabulary of computer systems. Most of the coding units devoted to this domain were from Book A and were found in all subcategories except H.1.13 (recognize the function of some basic communications technology, e.g., networks, distributed networks, modem or acoustics-coupler), H.1.4 (distinguish parallel and serial communications), and H.1.20 (distinguish analog and digital computing operations). Emphasis seems to be placed on the development of computer hardware since subcategory H.1.15 (recognize the rapid growth of computer hardware since 1940) earns a larger percentage of 28%. Textbook B covers very few instructional objectives in this area. Only 8 coding units were found in subcategories H.1.1, H.1.2, H.1.6, and H.1.11 involving physical components of computer systems. Both text materials neglect the objectives in subcategories H.1.13 and H.1.14 concerning communications technology, and subcategory H.1.20 involving the difference in operations of analog and digital computers. Possible explanations for the absence of subcategories H.1.13 and H.1.4 is that communications technology, like mainframes or networks, is still unavailable for classroom settings.
Table 5
Percentages of coding units devoted to instructional objectives in hardware domain classified according to the MECC analytical scheme.

<table>
<thead>
<tr>
<th>Content Category</th>
<th>Sub-category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Hardware</td>
<td>H.1.1</td>
<td>3</td>
<td>2.3</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>H.1.2</td>
<td>2</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>H.1.3</td>
<td>3</td>
<td>2.3</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.4</td>
<td>12</td>
<td>9.1</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.5</td>
<td>2</td>
<td>1.5</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.6</td>
<td>2</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>H.1.7</td>
<td>3</td>
<td>2.3</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.8</td>
<td>9</td>
<td>6.8</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.9</td>
<td>9</td>
<td>6.8</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.10</td>
<td>14</td>
<td>10.6</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.11</td>
<td>14</td>
<td>10.6</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>H.1.12</td>
<td>5</td>
<td>3.8</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.13</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.14</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.15</td>
<td>37</td>
<td>28.0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.16</td>
<td>8</td>
<td>6.1</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.17</td>
<td>2</td>
<td>1.5</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.18</td>
<td>3</td>
<td>2.3</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.19</td>
<td>4</td>
<td>3.0</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>H.1.20</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>132</td>
<td>100.0</td>
<td>8</td>
</tr>
</tbody>
</table>

An analysis of instructional objectives in the impact domain is presented in Table 6. This domain encompasses the social effects of computerization, including both the positive and negative impacts of computers on society. All coding units were from Book A only and 81.6% can be
Table 6
Percentages of coding units devoted to instructional objectives in the impact domain classified according to the MECC analytical scheme.

| Content Category | Sub-category | Book A | | Book B | | Composite |
|------------------|--------------|--------|--------|--------|--------|
|                  | f | % | f | % | f | % |
| Impact           |    |    |    |    |    |    |
| I.1.1            | 24 | 14.2 | - | - | 24 | 14.2 |
| I.1.2            | 24 | 14.2 | - | - | 24 | 14.2 |
| I.1.3            | 8  | 4.7 | - | - | 8  | 4.7  |
| I.1.4            | 5  | 3.0 | - | - | 5  | 3.0  |
| I.1.5            | -  | -  | - | - | -  | -    |
| I.1.6            | -  | -  | - | - | -  | -    |
| I.1.7            | -  | -  | - | - | -  | -    |
| I.1.8            | -  | -  | - | - | -  | -    |
| I.1.9            | -  | -  | - | - | -  | -    |
| I.1.10           | 2  | 1.2 | - | - | 2  | 1.2  |
| I.1.11           | 1  | 0.6 | - | - | 1  | 0.6  |
| I.1.12           | 1  | 0.6 | - | - | 1  | 0.6  |
| I.1.13           | 3  | 1.8 | - | - | 3  | 1.8  |
| I.2.1            | -  | -  | - | - | -  | -    |
| I.2.2            | 6  | 3.5 | - | - | 6  | 3.5  |
| I.2.3            | 2  | 1.2 | - | - | 2  | 1.2  |
| I.2.4            | 23 | 13.6| - | - | 23 | 13.6 |
| I.2.5            | 39 | 23.1| - | - | 39 | 23.1 |
| Others           | 31 | 18.3| - | - | 31 | 18.3 |
| Total            | 169| 100.0| - | - | 169| 100.0|

Identified as meeting some of the impact objectives. Within this portion, objectives that were considered significant in the advanced level were in subcategories I.2.5 (identify and evaluate the positive and negative consequences of computer use in specific situations), and I.2.4 (explain how
computers can be used to impact the distribution and use of economic, social and political power), and in the elementary level were in I.1.1 (describe some of the more common computer-related careers such as computer programmer, system analyst, computer scientist, computer operator, key operator), and I.1.2 (recognize that computers and computer-supported applications have a major impact on the way ordinary citizens live, work, and play). There were no instructional objectives found in I.1.5 (recognize that identification codes [numbers] and passwords are a primary means for restricting use of computer system, of computer programs, and of data file), I.1.6 (recognize that the procedures for detecting computer-based crimes are limited), I.1.7 (identify some advantages or disadvantages of a database containing personal information on a large number of people; the list might include value for research and potential for privacy invasion), I.1.8 (recognize several regulatory procedures, e.g., privilege to review one's own file and restrictions on use of universal identifiers, which help to insure the integrity of data files), I.1.9 (recognize that "privacy problems" are characteristics of large information files whether or not they are computerized), and I.2.1 (plan a strategy for tracing and correcting a computer-related error such as a billing error). The other 18.3% assigned to the "others" subcategory, deal with people in the historical development
of computers. The text materials failed to include an important facet of social impact of computerization as presented in subcategories I.1.5 through I.1.9 that concerned computer privacy.

An analysis of instructional objectives in the limitations category which focuses on developing a general sense of capabilities and limitations of computing machines is shown in Table 7. As in the impact domain, instructional objectives in limitations are only presented in Book A. The result indicates that the material places emphasis on subcategory L.1.6 (distinguish major difference between human capabilities and computer capabilities and omits the

Table 7
Percentages of coding units devoted to instructional objectives in limitations domain classified according to the MECC analytical scheme.

<table>
<thead>
<tr>
<th>Content Category</th>
<th>Sub-category</th>
<th>Book A</th>
<th></th>
<th>Book B</th>
<th></th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
<td>%</td>
<td>f</td>
<td>%</td>
</tr>
<tr>
<td>Limitations</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L.1.1</td>
<td>1</td>
<td>3.1</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>3.1</td>
</tr>
<tr>
<td>L.1.2</td>
<td>2</td>
<td>6.3</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>6.3</td>
</tr>
<tr>
<td>L.1.3</td>
<td>3</td>
<td>9.4</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>9.4</td>
</tr>
<tr>
<td>L.1.4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>L.1.5</td>
<td>1</td>
<td>3.1</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>3.1</td>
</tr>
<tr>
<td>L.1.6</td>
<td>25</td>
<td>78.1</td>
<td>-</td>
<td>-</td>
<td>25</td>
<td>78.1</td>
</tr>
<tr>
<td>Total</td>
<td>32</td>
<td>100.0</td>
<td>-</td>
<td>-</td>
<td>32</td>
<td>100.0</td>
</tr>
</tbody>
</table>
objective in subcategory L.1.4 (recognize that despite "artificial intelligence," computers cannot think in the way that we normally use the word "think").

Instructional objectives in the programming/algorithms domain which deal with the ability to read, modify, and construct algorithms and programs are shown in Table 8. In Textbook A, 47.5% of the coding units meet some of the MECC instructional objectives and 52.5% of the units were assigned to the "others" subcategory. Most of the classifiable units were in elementary levels, particularly in subcategory P.1.1 (recognize the definition of algorithm and that flowcharts and programs are alternative forms for expressing algorithms). Coding units in the "others" subcategory cover the knowledge of program development and the objectives are to enable students to understand the basic steps in developing a computer program i.e., problem analysis, program design, program writing, data entering, and program verification; analysis of a given problem in terms of output, input, and data processing; identification of the definition of flowchart symbols; and identification of major structures of flowcharting i.e., sequencing, selection, and repetition. In Textbook B, only about 14.4% were classified as meeting some categories of the MECC objectives from subcategories P.1.3 through P.2.4. A larger percentage of 85.6% of the coding units deal with syntaxes and constructs for programming in the BASIC language, which
### Table 8

Percentages of coding units devoted to instructional objectives in programming/algorithms domain classified according to the MECC analytical scheme.

<table>
<thead>
<tr>
<th>Content Category</th>
<th>Sub-category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Programming/Algorithms</td>
<td>P.1.1</td>
<td>66</td>
<td>30.4</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>P.1.2</td>
<td>19</td>
<td>8.8</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>P.1.3</td>
<td>4</td>
<td>1.8</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>P.1.4</td>
<td>6</td>
<td>2.8</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>P.2.1</td>
<td>8</td>
<td>3.7</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>P.2.3*</td>
<td>-</td>
<td>-</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>P.2.4</td>
<td>-</td>
<td>-</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>P.2.5</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Others</td>
<td></td>
<td>114</td>
<td>52.5</td>
<td>540</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>217</td>
<td>100.0</td>
<td>631</td>
</tr>
</tbody>
</table>

*Collapsed with subcategory P.2.2

include simple BASIC commands, statements, and functions. An even larger number of coding units from both textbooks could not be classified as meeting the MECC objectives and are prerequisites to all of those instructional objectives in this category. The text materials appear to cover all objectives except one in advanced subcategory P.2.5 (select an appropriate algorithm from a set of alternatives using criteria such as efficiency, elegance, and appropriateness).
The percentages of coding units representing instructional objectives in software and information processing category are illustrated in Table 9. This category includes vocabulary relevant to software, information processing, and data. Most of the 163 coding units

<table>
<thead>
<tr>
<th>Content Category</th>
<th>Sub-category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
<td>%</td>
</tr>
<tr>
<td><strong>Software and Information Processing</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S.1.1</td>
<td>27</td>
<td>16.6</td>
<td>3</td>
<td>11.1</td>
</tr>
<tr>
<td>S.1.2</td>
<td>51</td>
<td>31.3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.1.3</td>
<td>1</td>
<td>0.6</td>
<td>7</td>
<td>25.9</td>
</tr>
<tr>
<td>S.1.4</td>
<td>3</td>
<td>1.8</td>
<td>10</td>
<td>37.0</td>
</tr>
<tr>
<td>S.1.5</td>
<td>11</td>
<td>6.8</td>
<td>4</td>
<td>14.9</td>
</tr>
<tr>
<td>S.1.6</td>
<td>9</td>
<td>5.5</td>
<td>3</td>
<td>11.1</td>
</tr>
<tr>
<td>S.1.7</td>
<td>11</td>
<td>6.8</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.1.8</td>
<td>4</td>
<td>2.5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.1.9</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.1.10</td>
<td>14</td>
<td>8.6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.1.11</td>
<td>3</td>
<td>1.8</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.1.12</td>
<td>16</td>
<td>9.7</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.1.13</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.1.14</td>
<td>4</td>
<td>2.5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.2.1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.2.2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S.2.3</td>
<td>2</td>
<td>1.2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Others</td>
<td>7</td>
<td>4.3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>163</td>
<td>100.0</td>
<td>27</td>
<td>100.0</td>
</tr>
</tbody>
</table>
units were from Textbook A; 95.7% covered all subcategories except S.1.9, S.1.13, S.2.1, and S.2.2, and 4.3% were in the "others" subcategory. About half of the coding units in Book A represented subcategories S.1.2 (recognize that digital computers operate upon information which has been encoded in binary, the coding system which can utilize the base 2 number system), and S.1.4 (recognize that a computer gets instructions from a program written in a programming language by a person). A smaller portion of coding units in Textbook B covers some of the MECC instructional objectives in the elementary levels. The coding units distributed in descending order among subcategories are S.1.4, S.1.3 (recognize that a computer needs instructions to operate), S.1.5 (recognize that computer programs are set of sequential instructions which perform many tasks such as printing, sorting, calculating), S.1.6 (identify that the basic elements of a information processing system are input, processing, and output, in that order), and S.1.1 (identify that software refers to computer programs and includes operating systems, compilers, and user programs). On the whole, about 96.3% of the coding units can be classified as meeting some of the MECC instructional objectives. Most fell into the elementary level. Only 1% in subcategory S.2.3 (design an elementary coding system for a given application) were in an advanced level. None were found in subcategories S.1.9 (identify the fact that communication is
the transmission of information via coding message), S.1.13 (recognize that software refers to any nonpermanents sets of program whereas firmware refers to software that has been made physically permanent, S.2.1 (select and appropriate attribute for ordering of data for a particular task), and S.2.2. (design an elementary data structure for a given application that is, provide order for the data). The coding units that cannot be identified as meeting MECC objectives are devoted to identifying methods by which data can be entered into computers, either via keyboard or retrieval from secondary storage.

Shown in Table 10 is the classification of coding units in the usage domain which involves motor skills for sequencing and execution of tasks on computers or computer terminals. Only 16 coding units were found in Textbook A, representing only subcategory U.1.5 (enter, compile, and debug a simple stored program written in a "higher level" language). Of the 83 coding units in Book B, the highest percentage fell into subcategory U.1.5., the second highest percentage were classified in "others," the third were in U.1.3 (use system commands for an available computer system), and the last were in U.1.1 (connect a microcomputer or computer terminal to a power source and available storage units such as disks or tapes). No coding units were found in U.1.2 (interact with a computer in an online instructional learning situation involving drill and
Table 10
Percentage of coding units devoted to instructional objectives in usage domain classified according to the MECC analytical scheme.

<table>
<thead>
<tr>
<th>Content Category</th>
<th>Sub-category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Usage</td>
<td>U.1.1</td>
<td>1</td>
<td>100.0</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>U.1.2</td>
<td>-</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>U.1.3</td>
<td>19</td>
<td>22.9</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>U.1.4</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>U.1.5</td>
<td>16</td>
<td>100.0</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>Others</td>
<td>-</td>
<td>-</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td>16</td>
<td>100.0</td>
<td>83</td>
</tr>
</tbody>
</table>

dialogs), and U.1.4. (use program documentation to select and run library programs for specific tasks). Those in the "others" subcategory involved practice sessions on the keyboard. The greatest number of coding units in subcategory U.1.5 and the absence of units in subcategories U.1.2 and U.1.4 clearly demonstrates that the computer is generally limited to learning the program. It should be noted that only one coding unit, found in subcategory U.1.1, was less than what it was expected since instructional objectives in subcategories U.1.3, U.1.5, and "others" were actually associated with the physical use of computers that involve motor skills in connecting the computer system. In
addition, hands-on experience with computers was found in some other categories under programming/algorithms, and application, where students engage in programming. This may result from the rigidity imposed by the definition of this particular subcategory (U.1.1), and because the textbook did not include a means for evaluation of students’ actual experience with computers, especially outside the classroom. On the whole, a majority, or 76.8% of coding units represented some of the MECC objectives and the minority, or 23.2% represented the "others" category of objectives.

PART 2: To what extent do the text materials focus on each mode of instruction?

Summarized in Table 11 are the results from the classification of the coding units according to the HSDE analytical scheme, which explains how the information in the eight content domains in both textbooks were delivered to students based on the uses of computers. As seen from the data displayed, there are large differences between the two textbooks. Textbook A, with an emphasis on computer awareness, used only topic and tool modes in delivering the contents to students, while Textbook B, with an emphasis on programming, used only tutee and topic modes. Of the 873 coding units in Book A, 94% were in topic mode, and 6% were in tool mode. The pattern is different in Textbook B, where about 67% of the 811 coding units were in the tutee mode and the other 33% were in the topic mode. When considering both
Table 11

Percentages of coding units devoted to each instructional mode classified according to the HSDE analytical scheme.

<table>
<thead>
<tr>
<th>Instructional Mode</th>
<th>Book A</th>
<th></th>
<th>Book B</th>
<th></th>
<th>Composite</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td></td>
<td>f</td>
<td></td>
<td>%</td>
</tr>
<tr>
<td>Topic</td>
<td>818</td>
<td>93.7</td>
<td>268</td>
<td>33.1</td>
<td>1086</td>
<td>64.5</td>
</tr>
<tr>
<td>Tutor</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Tutee</td>
<td>-</td>
<td>-</td>
<td>543</td>
<td>66.9</td>
<td>543</td>
<td>32.2</td>
</tr>
<tr>
<td>Tool</td>
<td>55</td>
<td>6.3</td>
<td>-</td>
<td>-</td>
<td>55</td>
<td>3.3</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>873</td>
<td>100.0</td>
<td>811</td>
<td>100.0</td>
<td>1684</td>
<td>100.0</td>
</tr>
</tbody>
</table>

textbooks, topic mode contained the greatest number of coding units, 64.5%, followed by tutee mode, 32.2%, and tool mode, 3.3%. No coding units were identified as using the tutor mode of instruction. A plausible explanation for the relatively low use of computers in the tool mode or the absence of computer use in the tutor mode is largely due to the lack of appropriate software, which is an essential ingredient in both tutor and tool modes.

Further analysis addressed how the content in all categories is presented across the program in each mode of instruction. Tables 12 through 14 present the summary for the topic, tutee, and tool modes.

Table 12 is a listing of content categories, along with the number of coding units found in the topic mode with
Table 12

Percentages of coding units devoted to topic mode by content category classified according to the HSDE analytical scheme.

<table>
<thead>
<tr>
<th>Instructional Mode</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Topic Mode</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>144</td>
<td>17.6</td>
<td>18</td>
</tr>
<tr>
<td>Hardware</td>
<td>130</td>
<td>15.9</td>
<td>8</td>
</tr>
<tr>
<td>Impact</td>
<td>169</td>
<td>20.7</td>
<td>-</td>
</tr>
<tr>
<td>Limitations</td>
<td>32</td>
<td>3.9</td>
<td>-</td>
</tr>
<tr>
<td>Programming/Algorithms</td>
<td>217</td>
<td>26.5</td>
<td>186</td>
</tr>
<tr>
<td>Software &amp; Information Processing</td>
<td>126</td>
<td>15.4</td>
<td>22</td>
</tr>
<tr>
<td>Usage</td>
<td>-</td>
<td>-</td>
<td>34</td>
</tr>
<tr>
<td>Values and Feelings</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>818</strong></td>
<td><strong>100.0</strong></td>
<td><strong>268</strong></td>
</tr>
</tbody>
</table>

which computer is not directly used. Of the 818 units in Textbook A, the content in all areas, except usage, and values and feelings was presented to students via the topic mode, depending most often on lecture, discussion, and demonstration teaching methods. Units in Book B, containing 268 coding units, were in the areas of application, hardware, programming/algorithms, software/information processing, and usage. As indicated by the largest percentage in both textbooks, the topic mode was used more in programming/algorithms than in other content categories,
while the distribution for application, hardware, impact, and software and information processing varied slightly, and provided a very small number of units for usage and limitations.

The use of tutee mode in which students or teachers communicate with the computer by programming it is shown in Table 13. Only the coding units from Textbook B in the

Table 13

Percentages of coding units devoted to tutee mode by content category classified according to the HSDE analytical scheme.

<table>
<thead>
<tr>
<th>Instructional Mode Content Category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tutee</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>-</td>
<td>-</td>
<td>44 8.1</td>
</tr>
<tr>
<td>Hardware</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Impact</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Limitations</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Programming/Algorithms</td>
<td>-</td>
<td>-</td>
<td>445 81.9</td>
</tr>
<tr>
<td>Software/Information</td>
<td>-</td>
<td>-</td>
<td>5 1.0</td>
</tr>
<tr>
<td>Processing</td>
<td>-</td>
<td>-</td>
<td>5 1.0</td>
</tr>
<tr>
<td>Usage</td>
<td>-</td>
<td>-</td>
<td>49 9.0</td>
</tr>
<tr>
<td>Values and Feelings</td>
<td>-</td>
<td>-</td>
<td>49 9.0</td>
</tr>
<tr>
<td>Total</td>
<td>543 100.0</td>
<td>543 100.0</td>
<td></td>
</tr>
</tbody>
</table>

areas of application, programming/algorithms, software and information processing, and usage were classified as using this mode of instruction. As expected, Book B which was devoted primarily to programming, frequently used the tutee
mode in programming/algorithms, 82%, while usage, application, and software and information processing combined earned 18%.

Presented in Table 14 are the number of coding units which were classified as using the tool mode, in which the computer functions as an instructional aid providing service to both teachers and students. Only a small number of coding units were found in Textbook A as using the computer as a tool. Of the 55 coding units, 67.3% were in the areas of software and information processing, 29.1% were in usage, and 3.6% were in hardware. The computers were used to

<table>
<thead>
<tr>
<th>Instructional Mode</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Category</td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Tool Mode</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Hardware</td>
<td>2</td>
<td>3.6</td>
<td>-</td>
</tr>
<tr>
<td>Impact</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Limitations</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Programming/Algorithm</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Software/Information</td>
<td>37</td>
<td>67.3</td>
<td>-</td>
</tr>
<tr>
<td>Processing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usage</td>
<td>16</td>
<td>29.1</td>
<td>-</td>
</tr>
<tr>
<td>Values and Feelings</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td>55</td>
<td>100.0</td>
<td>-</td>
</tr>
</tbody>
</table>
provide service to teachers, as an aid to generating curriculum material. An example of using a computer as a tool found in software and information processing, is the demonstration of the program illustrating ideas and concepts on binary number system to the student.

PART 3: To what extent do the text material questions focus on higher cognitive levels?

The question addressed in this part involves classification of all review questions and exercises in the text materials according to Bloom's Taxonomy of Educational Objective, Cognitive Domain. These data and the summaries of the results of the classification of the coding units into the six levels of Bloom's taxonomy are presented in Table 15. Perhaps the most surprising findings were that the two books, with different approaches, were so similar in their neglect of the two highest levels of cognitive skills, synthesis and evaluation. In Textbook A, the number of questions was equally classified as low and high cognitive level groups. Of the 94 coding units examined, 50% were in the knowledge level, and the 50% classified in the higher-thinking operation group included comprehension, application, and analysis levels. Only 1 analysis question was found at the highest level of thinking skill. Of the 163 question units in Textbook B, with a programming emphasis, 37.4% were judged as low thinking process questions and 62.6% were considered higher thinking process
Table 15

Percentages of question units devoted to each cognitive level classified according to Bloom's cognitive level.

<table>
<thead>
<tr>
<th>Cognitive Level</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Knowledge</td>
<td>47</td>
<td>50.0</td>
<td>61</td>
</tr>
<tr>
<td>Comprehension</td>
<td>31</td>
<td>33.0</td>
<td>26</td>
</tr>
<tr>
<td>Application</td>
<td>15</td>
<td>15.9</td>
<td>62</td>
</tr>
<tr>
<td>Analysis</td>
<td>1</td>
<td>1.1</td>
<td>14</td>
</tr>
<tr>
<td>Synthesis</td>
<td></td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>Evaluation</td>
<td></td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td>94</td>
<td>100.0</td>
<td>163</td>
</tr>
</tbody>
</table>

questions. The larger percentage of higher level questions fell primarily in the application category, and the smaller percentage in the comprehension and analysis. Of all 257 question units from both textbooks, 42% of the questions used concerned low thinking skills and 58% concerned higher thinking skills, from comprehension to analysis. In all, the text materials reflected fairly frequent use of higher level questions. A detailed analysis of how questions in each cognitive level were used among content categories is presented in Tables 16 through 19.

Illustrated in Table 16 is the use of questions in the knowledge level that requires only rote memory of fact, terminologies, or routine operations of problems previously
learned. Of the 108 question units classified in this low
cognitive level, 47 units were from Book A and 61 units were

Table 16

Percentages of question units devoted to knowledge
level by content category classified according to
Bloom's taxonomy.

<table>
<thead>
<tr>
<th>Cognitive Level Content Category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Knowledge Level</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>2</td>
<td>4.3</td>
<td>-</td>
</tr>
<tr>
<td>Hardware</td>
<td>20</td>
<td>42.6</td>
<td>-</td>
</tr>
<tr>
<td>Impact</td>
<td>12</td>
<td>25.5</td>
<td>-</td>
</tr>
<tr>
<td>Limitations</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Programming/Algorithm</td>
<td>6</td>
<td>12.8</td>
<td>61 100.0</td>
</tr>
<tr>
<td>Software/Information Processing</td>
<td>7</td>
<td>14.8</td>
<td>-</td>
</tr>
<tr>
<td>Usage</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Values and Feelings</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td>47 100.0</td>
<td>61 100.0</td>
<td>108 100.0</td>
</tr>
</tbody>
</table>

from Book B. The knowledge questions in Book A were used
most in hardware (42.6%), followed by impact (25.5%),
software and information processing (14.8%), programming/algorithms (12.8%), and application (4.3%). All
61 question units in Textbook B were in the area of
programming/algorithms.

The frequency of use of questions in the comprehension
category which requires the student to understand certain
information, and be able to translate, interpolate, and extrapolate that particular information is shown in Table 17. Of the 47 coding units in Textbook A, most were in software and information processing and hardware categories, and a very small number of units were in the application.

Table 17
Percentages of question units devoted to comprehension level by content category classified according to Bloom's taxonomy.

<table>
<thead>
<tr>
<th>Cognitive Level</th>
<th>Content Category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comprehension</td>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Level</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Application</td>
<td>4</td>
<td>12.9</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Hardware</td>
<td>7</td>
<td>22.5</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Impact</td>
<td>3</td>
<td>9.6</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Limitations</td>
<td>3</td>
<td>9.6</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Programming/Algorithms</td>
<td>1</td>
<td>3.2</td>
<td>26 100.0</td>
</tr>
<tr>
<td></td>
<td>Software/Information</td>
<td>13</td>
<td>41.9</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Processing</td>
<td></td>
<td></td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>usage</td>
<td></td>
<td></td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Values and Feelings</td>
<td></td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>31</td>
<td>100.0</td>
<td>26 100.0</td>
</tr>
</tbody>
</table>

programming/algorithms, impact, and limitation categories. All of the 26 question units in Book B were classified in one single category, programming/algorithms. Of the 57 coding units, programming/algorithms, and software and information processing contain the greatest number of
comprehension questions than other content categories.

The frequency with which questions that demand application of the knowledge to solving problems were used is shown in Table 18. Fifteen question units in Book A were used in four content categories: impact, limitations, programming/algorithms, and software and information processing. The greatest number of question units, 10, were found in the software and information processing category.

In Textbook B, most of the 62 questions were used in the programming/algorithms category. Only 8 and 2 question

Table 18

Percentages of question units devoted to application level by content category classified according to Bloom's taxonomy.

<table>
<thead>
<tr>
<th>Cognitive Level Content Category</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Application Level</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>-</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>Hardware</td>
<td>1</td>
<td>6.7</td>
<td>-</td>
</tr>
<tr>
<td>Impact</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Limitations</td>
<td>1</td>
<td>6.7</td>
<td>-</td>
</tr>
<tr>
<td>Programming/Algorithms</td>
<td>3</td>
<td>20.0</td>
<td>52</td>
</tr>
<tr>
<td>Software/Information Processing</td>
<td>10</td>
<td>66.6</td>
<td>-</td>
</tr>
<tr>
<td>Usage</td>
<td>-</td>
<td>-</td>
<td>8</td>
</tr>
<tr>
<td>Values and Feelings</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td>15</td>
<td>100.0</td>
<td>62</td>
</tr>
</tbody>
</table>
units were used in the usage and application categories, respectively. The use of application questions in the total program were found most often in the programming/algorithms category, but none were found in the impact or values and feelings categories.

Illustrated in Table 19 is the use of analytic thinking questions that involve an ability to solve non-routine problems. Both textbooks appeared to give little importance to the use of analytic questions. Only 1 question unit in Textbook A dealt with this higher mental operation process.

Table 19

Percentages of question units devoted to analysis level by content category classified according to Bloom's taxonomy.

<table>
<thead>
<tr>
<th>Cognitive Level</th>
<th>Book A</th>
<th>Book B</th>
<th>Composite</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content Category</td>
<td>f</td>
<td>%</td>
<td>f</td>
</tr>
<tr>
<td>Analysis Level</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>-</td>
<td>-</td>
<td>2 14.3</td>
</tr>
<tr>
<td>Hardware</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Impact</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Limitations</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Programming/Algorithm</td>
<td>1 100.0</td>
<td>12 85.7</td>
<td>13 86.7</td>
</tr>
<tr>
<td>Software/Information Processing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Usage</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Values and Feelings</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Total</td>
<td>1 100.0</td>
<td>14 100.0</td>
<td>15 100.0</td>
</tr>
</tbody>
</table>
in the programming/algorithms. Textbook B had more analytic questions than Book A, but contained only 12 question units in the programming/algorithms and 2 question units in the usage category.

Based on the data presented in Tables 16 through 19, it was found that, of all 257 question units within the computer literacy program, questions were used most often in programming/algorithms (63%), and in software and information processing (11.7%), hardware (10.9%), impact (5.84%), application (3.9%), usage (3.1%), and limitations (1.6%) in descending order by frequency.
CHAPTER V

SUMMARY, CONCLUSIONS, AND IMPLICATIONS

Summary

As previously stated, this study was an attempt to analyze the national computer literacy program, as reflected in computer literacy text materials, by using a content analysis technique. The specific purpose of the study concerns the instructional objectives, the specific modes of instructions, and the use of questions in text materials. Knowing the extent to which text materials focus on each computer literacy objective domain can serve as a basis for selecting intended learning outcomes. Understanding the extent to which text materials focus on each specific mode of instruction can provide information on how extensively computers are used in learning situations. Determining the extent to which text material questions focus on higher cognitive levels provides information on how questions in the text materials are used to enhance students' cognitive skills. Thus, the findings obtained from this study can contribute to the improvement of the existing national computer literacy program. This study contributes to the revision of the computer literacy curriculum materials, the improvement of computer teaching effectiveness, and the development of computer literacy programs for both
preservice and inservice teachers.

The findings of a content analysis of the IPST computer literacy text materials can be summarized as follow:

1. To what extent do text materials focus on each computer literacy objective domain?

The IPST computer literacy program which is comprised of Textbook A and Textbook B, includes 94.1% of cognitive objectives, application, hardware, impact, limitations, programming/algorithms, and software and information processing; and 5.9% of psychomotor objectives, usage. The program does not address any of the affective instructional objectives, values and feelings.

An analysis by content category revealed that all objectives addressed in Textbook A fell into the categories of programming/algorithms (24.9%), impact (19.4%), software and information processing (18.6%), application (16.5%), hardware (15.1%), limitations (3.7%), and usage (1.8%). The objectives in Textbook B fell into the categories of programming/algorithms (77.8%), usage (10.2%), application (7.6%), software and information processing (3.3%), and hardware (1%). The total program focused, in descending order, on programming/algorithms (50.4%), application (12.2%), software and information processing (11.3%), impact (10%), hardware (8.3%), usage (5.9%), and limitations (1.9%).
Further examination of the computer literacy objectives in all content dimensions indicated that the program does not precisely conform to instructional objectives in all subcategories according to the MECC classifications. The average adoption of the MECC objectives was 86.2% in Textbook A, and 32.6% in Textbook B. Among the adopted objectives within each content category, the IPST program places most emphasis on subcategories A.2.4 (recognize some general criteria for computer usage for a particular task); H.1.15 (recognize the rapid growth of computer hardware since the 1940s); I.2.5 (identify and evaluate the positive and negative consequences of computer use in specific situations); L.1.6 (distinguish major differences between human capabilities and computer capabilities); P.1.1 (recognize the definition of algorithm and that flowcharts and programs are alternative forms for expressing algorithms); S.1.2 (recognize that digital computers operate upon information which has been encoded in binary, the coding system which can utilize the base 2 number system); and U.1.5 (enter, compile, and debug a simple stored program written in a "higher level" language).

The IPST program does not include instructional objectives in the following subcategories in its program: A.1.4 (recognize the definition of and some advantages of computer simulation); H.1.13 (recognize the function of some basic communications technology); H.1.14 (distinguish
parallel and serial communications); H.1.20 (distinguish analog and digital computing operations); I.1.5 (recognize that identification codes and passwords are a primary means for restricting use of computer systems, computer programs, and data files); I.1.6 (recognize that the procedures detecting computer-based crimes are limited); I.1.7 (identify some advantages or disadvantages of a data base containing personal information on a large number of people—the list might include value for research and potential for privacy invasion); I.1.8 (recognize several regulatory procedures which help to insure the integrity of personal data files); I.1.9 (recognize that "privacy problems" are characteristic of large information files whether or not they are computerized); I.2.1 (plan a strategy for tracing and correcting a computer-related error such as a billing error); L.1.4 (recognize that despite "artificial intelligence," computers cannot think in the way the word "think" is normally used); P.2.5 (select an appropriate algorithm from a set of alternatives using criteria such as efficiency, elegance, and appropriateness); S.1.9 (identify the fact that communication is the transmission of information via coded messages); S.1.13 (recognize that software refers to any nonpermanent set of programs whereas firmware refers to software that has been made physically permanent); S.2.1 (select an appropriate attribute for ordering of data for a particular task); S.2.2
(design an elementary data structure for a given application); U.1.2 (interact with a computer in an on-line instructional learning situation involving drill and dialogs); U.1.4 (use program documentation to select and run library program written in a "higher level" language); V.1.1 (does not feel fear, anxiety, or intimidation from computer experiences); V.1.2 (feels confident about the ability to use and control computers); V.1.3 (enjoys and desires work or play with computers, especially computer-assisted learning); V.1.4 (describes past experiences with computers with positive-affect words like fun, exciting, challenging, etc.); V.1.5 (given an opportunity, spends free time using a computer); V.2.1 (values the potential role of computers in meeting societal and instructional needs); V.2.2 (values efficient information processing provided that it does not neglect accuracy, the protection of individual rights, or societal needs); V.2.3 (values computerization of routine tasks so long as it frees people to engage in other activities and is not done as an end in itself); and V.2.4. (values increased communication and availability of information made possible through computer use provided that it does not violate personal rights to privacy and accuracy of personal data).

Objectives that did not conform to the MECC objectives and were assigned to the "others" subcategories included 13.8% in Textbook A within the impact,
programming/algorithms, and software and information processing; and 67.8% in Textbook B, within the programming/algorithms category. These objectives are established to enable students to recognize the people who have played major roles in the historical development of computers (impact); to identify methods of entering data to computers (software and information processing); and to understand the basic steps in developing a computer program, to analyze a given problem in terms of output, input, and data processing, to identify flowchart symbols and the major structure of flowcharting, as well as to recognize the syntaxes and constructs for programming in the BASIC language which include simple BASIC commands, statements, and functions (programming/algorithms).

2. To what extent do text materials focus on each mode of instruction?

Reexamination of the 1,684 categorized units in all content categories indicates that the IPST computer literacy program used only three modes of instructions: topic tutee, and tool. The program emphasized topic (64.5%) most often, tutee (32.5%) second, and the tool mode (3.3%) least often. The use of tutor mode was not found at all. There was considerable difference in the methods for using computers to deliver content to students between the two books. Most of the content in Textbook A is presented to students using the topic mode (93.7%). Only 3.3% of the
content in Textbook A dealt with the tool mode. In Textbook B, content is presented to students using the topic and tutee modes with percentages of 33.1% and 66.9%, respectively.

When considering each mode of instruction, the topic mode was used in all areas except values and feelings. The percentage of distributions in these content categories, in descending order, are 26.5% programming/algorithms, 20.7% impact, 17.6% application, 15.9% hardware, 15.4% software and information processing, and 3.9% limitations. The tutee mode was used frequently in the area of programming/algorithms, 81.9%; less in the areas of usage, 9%, and application, 8.1%; and seldom in the area of software and information processing, 1%. Of the smallest number of content units in the tool mode, 67.3% were in software and information processing, 29.1% were in usage, and 3.6% were in hardware.

3. To what extent do text material questions focus on higher cognitive level?

The results from the classification of the 257 question units using Bloom's taxonomy indicate that 42% of the questions focus on the lowest cognitive skill, knowledge; and that 58% of the questions focus on higher cognitive skills, 22.2% comprehension, 30% application, and 5.5% analysis; The IPST text materials do not have questions that focus on the two highest cognitive skills, synthesis and
The percentages of the question units devoted to each cognitive levels within textbooks varies substantially. The cognitive level of questions in Textbook A were 50% knowledge, 33% comprehension, 15.9% application, and 1.1% analysis; while the levels of questions in Textbook B were 37.4% knowledge, 16% comprehension, 38% application, and 8.6% analysis.

Further analysis of each type of question by content category across the two books revealed that knowledge questions were used in programming/algorithms (62%), hardware (18.5%), impact (11.1%), software and information processing (6.5%), and application (1.9%). The comprehension questions were used in programming/algorithms (47.3%), software and information processing (22.8%), hardware (12.3%), application (7%), impact (5.3%), and limitations (5.3%). The application questions were used in programming/algorithms (71.4%), software and information processing (13%), usage (10.4%), application (2.6%), hardware (1%), and limitations (1%). The analysis questions were used most in programming/algorithms (86.7%), and application (13.3%).
Discussion

The IPST computer literacy program, as reflected from the text materials, include instructional objectives in cognitive, psychomotor, but affective components. The program appears to focus most on cognitive aspects of computer learning, almost to the exclusion of any other aspect of learning. This finding is consistent with that of most research which has found that most textbooks concern only cognitive gains and tend to overlook affective or psychomotor concerns.

With respect to cognitive learning, instructional objectives in the programming/algorithms content category received greatest emphasis while those in the limitations category received least emphasis. This can be explained by the fact that an awareness of the capabilities and limitations of computing machines can be developed through "hands-on" experience with computers. It is possible that students are expected to gain this "hands-on" experience while they are engaged in programming.

The relatively low emphasis placed on usage objectives in the psychomotor component may be partially caused by the failure to evaluate the uses of computers when the students perform the task in other content categories (e.g., programming/algorithms, and application), due to the rigidity imposed by the definition of the analytical scheme.
The text materials failed to include some of the instructional objectives addressed in the MECC computer literacy classification. The reason for this failure is due largely to hardware and software constraints. In developing countries such as Thailand, networking and telecommunications are still unaffordable in most classroom settings. For this reason, the use of computers cannot be extended to the access of large data bases or electronic bulletin board systems. The instructional objectives concerning communications technology, including related issues such as computer crime or computer privacy are not likely to be addressed in the program. Example of these objectives are addressed in subcategories H.1.13, H.1.14, S.1.9, I.1.6, I.1.7, I.1.8, and I.1.9.

In general, learning still takes place through the passive learning mode, or topic mode. Microcomputers are mainly used for teaching programming. The use of computers as a tool is not substantial. Computers are only used as an aid to the teacher to generate a few curriculum materials, not to provide services to students or to accomplish tasks. Essentially, the IPST computer literacy program uses only two modes of instruction in conveying course content to the students--topic and tutee modes. The lack of appropriate software limits the use of computers to tutoring students or to using as a tool because software is the essential ingredient to the use of computer in the tutor and tool
modes. Another factor that must be taken into account is that teachers have little knowledge of how to use the computers.

The questions and exercises in both textbooks served all cognitive objectives and were found to place more emphasis on higher intellectual process than knowledge acquisition. The majority of these higher cognitive questions emphasized the application of the knowledge learned. However, among these questions, none demand students to engage in synthetic or evaluative thinking. In light of the objectives held in this computer literacy program to foster critical thinking in students, and to the extent that the four categories of the taxonomy from application to evaluation correspond to the critical thinking process, the use of questions or exercises thus enhances students' cognitive thinking, and may partly contribute to the achievement of this objective.

Conclusions

Based on the findings of this study, the following conclusions seems warranted:

1. The program focuses on the cognitive aspects of computer learning, almost to the exclusion of affective or psychomotor learning.

2. In general, the principal mode of instruction is through the passive or topic learning mode.
3. The text materials focus on cognitive skills higher than the knowledge level, but may need revision to include more attention to higher order cognitive skills such as synthesis and evaluation.

Implications of the Study

The affective outcomes, values and feelings are particularly important objectives of computer instruction. Students must develop or maintain strong positive feelings toward computers. The disappearance of the values and feelings instructional objectives, does not imply that they are completely absent from the program. The failure to give instructional emphasis to affective outcomes is due to the difficulty in evaluating affective objectives because, unlike most cognitive objectives, they cannot be attained in the relatively short instructional period, and therefore cannot be evaluate in the school setting (Bloom, Hastings, & Madaus, 1971, pp. 226-227).

The relatively low emphasis on objectives in limitations and usage content categories and the absence of objectives in the values and feelings category suggest that instructional efforts should be launched to teach students about these facets of computer literacy. Extensive use of computers in all content categories should be fostered. Involving students in computer activities can provide hands-on experiences with computers and may include learning to turn the machine on and off, learning to operate disk
drives, and memorizing and using commands to load and run a program. As a result, students feel comfortable working with the computer system. By working with the machine, students develop a feeling, not only for how information is stored, retrieved, and possibly pirated, damaged, or destroyed, but also for the capabilities and limitations of the technology (Bullough & Beatty, 1987, p. 71; California State Department of Education, 1985, p. 19).

Hardware and software constraints have a great influence on course content, instructional objectives and modes of computer use in instruction. Microcomputers should be used far more than the teaching of programming. The uses of computers in the tutor or tool modes can improve and enrich classroom learning, and neither requires students or teachers to learn much about computers. However, in a developing country such as Thailand, where resources are extremely limited, the use of computers to tutor students may not be feasible. Most CAI software, tutorials, or drill and practices are text-intensive and make numerous assumptions about students' prior knowledge, culture, and vocabulary as well as the topics in the curriculum and their sequence. Simulations are often very specific and can be applied at only one point in a course. If schools wishes to use narrowly-focused software, a large number of individual programs must be reviewed, acquired, and managed in order to make a significant impact on instruction. This is a costly
undertaking which is beyond the resources of most schools (Tinker, 1987, pp. 119-120). Using software tools may be the most promising mode of instructional use of computers. Educational tool software such as word processors, data base management systems or spreadsheets offer the possibility of good pedagogy at low cost. In contrast to CAI software, they can be used in many contexts. A relatively small number of tools will satisfy the needs of many students in many courses. Because only a few tools are needed, teacher training requirements are simplified, and implementation of more than one computer is conceivable. Their limited use of text makes tool software easy to translate or to use without translation (Tinker, 1987, pp. 122).

Concerning the use of questions in the text materials, it is not clear what balance between exercises in knowledge acquisition and the higher cognitive process is appropriate in computer literacy courses. The issue needs careful examination and resolution by responsible educators. Once standards have been ascertained, teachers, and text materials authors should be made cognizant of them.

Recommendations for Further Research

Even though affective outcomes are complex and more difficult to evaluate than cognitive outcomes, attention must be paid to them. Excluding an affective aspect of computer education, values and feelings, will lead to an over-emphasis of verbal-conceptual instruction, and thus
provide no evidence on which to base modifications of its curriculum and pedagogical methods. Textbooks should include techniques with which to measure student outcome in the affective domain. A systematic effort to collect evidence of growth in these outcomes is strongly recommended. Empirical studies should be conducted to determine what types of appropriate learning experiences should be provided for students in order to develop the desired affective behaviors, how time spent at the computer affects student learning, and how are attitudes of students toward learning are affected by computers.

Considerable research and developments must focus on the use of computers in all aspects of education, including their effect on students' learning behavior and the changed nature of interactions in the classroom, what type of software works best, the impact of working with computers on students' learning process, whether or not the use of microcomputers for learning increases problem-solving abilities, how teachers should be prepared to teach with computers, etc.

Additional studies involving the use of questions in the textbooks should be conducted in order to explore the appropriate distribution between knowledge questions and higher cognitive questions, and the extent to which questions in each of the higher cognitive levels contribute to critical thinking and problem-solving skills.
APPENDIX A

COURSE CONTENT AND INSTRUCTIONAL OBJECTIVES OF
THE IPST COMPUTER LITERACY TEXT MATERIALS
COURSE CONTENT AND INSTRUCTIONAL OBJECTIVES OF
THE IPST COMPUTER LITERACY TEXT MATERIALS

Course 031: Introduction to Computers

Chapter One: The Roles of Computers in Daily Life

Content:
- Importance of computers
- Specific uses of computers and computers supported applications
- Computer personnel
- Definition of computers
- Evolutions of computers
- Generations of computers
- Kinds of computers

Objectives:
- Identify the roles and importance of computers in the present society.
- Trace the historical development of computers and describe characteristics of computers in each generation.
- Recognize the people who play major roles to the development of the computers.
- Identify different kinds of computers i.e., general-purpose computers, special-purpose computers, mainframes, minicomputers, microcomputers.

Chapter Two: Computer Systems

Content:
- Hardware
- Software
  - How hardware and software function in a computer system.

Objectives:
- Identify major components of the computer systems
- Identify major parts and functions of hardware i.e., memory, input, output, CPU.
- Identify different kinds of software i.e., operating system, programming language translator, utilities and application programs.
Chapter Three: Data

Content:
- Definition of data
- Types and characteristics of data
- Binary codes
- Data management
- How to input data to a computer

Objectives:
- Define the terms data and information, and give their examples.
- List types of data (numeric data and character data) and describe their characteristics.
- Identify types of data that can be used with computers from the given data.
- Define binary codes.
- Use binary code tables to code and decode given data.
- Describe terms related to data management, i.e., character, field, record, file.
- Describe how computers store data in their memory.
- Identify how data can be entered to computers and give examples of secondary storage and input device.

Chapter Four: Data Processing

Content:
- Definition of data processing
- Basic elements of data processing
- How to process data

Objectives:
- Describe the term data processing.
- Identify basic elements in data processing.
- Describe the differences between manual and machine data processing.

Chapter Five: Program and Program Development

Content:
- Computer programs
- Program development
- Problem analysis
- Program design
- Solving the problem through job analysis, flowchart and pseudocodes

Objectives:
- Describe how a computer operates by human instructions.
- List and describe the steps in developing a computer program (i.e., problem analysis, program design, program writing, data entering, and program verification).
- Analyze a given problem in terms of output, input, and data processing.
Identify and define flowchart symbols.
Identify major structures of flowcharting (i.e., sequencing, selection, and repetition).
Plan a program using flowchart to solve a given problem.
Describe briefly the meaning of pseudocode.
Identify characteristics of pseudocode structures (i.e., sequencing, selection, and repetition).
Plan a program using pseudocode to solve a given problem.
Solve a given problem by analyzing the problem through the use of flowcharts or pseudocode in a top-down structured design.

Chapter Six: Computer Language

Content: Machine language
Low-level language
High-level language
Functions of compiler and interpreter

Objectives: Discriminate between characteristics of different kinds of computer languages.
Give the names of higher-level languages.
Describe the characteristics of jobs appropriate to each kind of higher-level language.

Chapter Seven: Present and Future Uses of Computer

Content: Present uses of computers
Future uses of computers

Objectives: Give examples of jobs that use computers.
State the kinds of computers used in processing data at present.
Describe and list the steps in preparing jobs for computer uses.
List the types of computer programs used in data processing.
State the differences between using pre-existing package software and self-developed programs.
Give examples of computer uses in the future.
Give examples of the social impacts of computer uses.
Give examples of disadvantages and advantages of computer uses.
Chapter One: Introduction

Content: Historical development of BASIC language
Microcomputers

Objectives: Describe the development of BASIC language and its advantage.
Describe how a computer works, and how to use it.

Chapter Two: System Commands

Content: Definitions of system commands
First session at the keyboard
Entering a programming line
Making corrections on an entered programming line

Objectives: Use certain system commands i.e., HOME, NEW, LIST, RUN.
Make corrections by retyping or inserting an entered programming line.

Chapter Three: Elements of BASIC Commands

Content: Constants
Variables
Arithmetics/Logic operations
Expressions
Order in which operations are performed

Objectives: State the differences between a constant and a variable.
Given a set of constants and variables, identify which is string constant, numeric constant, logical constant, string variable, and numeric variable (integer or real).
Identify and describe the meanings of arithmetic/logic operation symbols.
Identify the order in which operations are performed.
Define the term expression and give its example.
Chapter Four: LET, PRINT, and REM Statements

Content: LET Statement
         PRINT Statement
         REM Statement

Objectives: Read and describe the program using LET, PRINT, and REM statements.
            Write a program using LET, PRINT, and REM statements.

Chapter Five: INPUT, READ, and RESTORE Statements

Content: INPUT statement
         READ statement
         RESTORE statement

Objectives: Describe how a program using INPUT, READ/DATA and RESTORE statements works.
            Identify types of variables that can be used with INPUT, READ/DATA statements.
            Write a program using INPUT, READ/DATA, and RESTORE statements.
            Identify which program should use INPUT statement or READ/DATA statements.

Chapter Six: FOR/NEXT Statement

Content: FOR/NEXT statement
         Nested FOR/NEXT loops
         Rules for writing FOR/NEXT loop

Objectives: Describe how a program using FOR/NEXT statements works.
            Write a program using FOR/NEXT loop.
            Describe how a program using FOR/NEXT statements in the nested loops works.
            Write a program using FOR/NEXT nested loops.

Chapter Seven: GOTO and IF/THEN Statements

Content: GOTO statement
         IF/THEN statement

Objectives: Describe how a program using GOTO, IF/THEN, and IF/THEN/ELSE statements work.
            Write a program using GOTO, IF/THEN, and IF/THEN/ELSE statements.
Chapter Eight: GOSUB/RETURN Statements

Content: Main program and subroutine
GOSUB/RETURN statements

Objectives: Describe how a program using GOSUB/RETURN statements works.
Write a program using GOSUB/RETURN statements.

Chapter Nine: ON/GOTO and ON/GOSUB Statements

Content: ON/GOTO statement
ON/GOSUB statement

Objectives: Describe how a program using ON/GOTO and ON/GOSUB statements works.
Write a program using ON/GOTO and ON/GOSUB statements.
Compare the differences between the program using ON/GOTO, and ON/GOSUB statements.

Chapter Ten: Fundamental BASIC Functions

Content: Arithmetic functions
Using arithmetics functions
String functions
Using string functions

Objectives: Define and describes how to use arithmetics and string functions.
Write a program using arithmetics and string functions.

Chapter Eleven: BASIC Programms and Their Applications

Content: Introduction to BASIC programming applications
Examples of BASIC application programs

Objectives: Describe the steps in developing the program.
Write the steps in solving a given problem including flowchart that problem.
Write an application program for a simple given task.
Identify and correct errors of a given program.
APPENDIX B

THE MECC COMPUTER LITERACY OBJECTIVES
THE MECC COMPUTER LITERACY OBJECTIVES

APPLICATIONS (A): This domain covers the multitude of social and organizational areas into which computers have been integrated. It also covers the general considerations for applying computers to new situations.

A.1.1 Recognize that computers and computer-supported applications are used in a wide variety of ways to assist individuals, groups and institutions.

A.1.2 Recognize specific uses of computers or computer-supported applications in many of the following areas:

a. medicine and health care
b. law enforcement and criminal justice
c. education and training
d. engineering and design
e. business and banking
f. the office
g. transportation and traffic control
h. defense system
i. weather prediction
j. recreation
k. government and politics
l. the home
m. the library
n. manufacturing
o. the creative arts
p. publishing
q. public utilities—telephone and power plants

A.1.3 Recognize that the following are common types of computer applications, and be able to give an example of each type.

a. information storage and retrieval—record keeping, data bases, etc.
b. simulation and modeling
c. process/machine control—robotics
d. computation (numerical/statistical analysis of data)
e. data processing
f. word processing
g. graphics
h. speech synthesis
i. artificial intelligence
A.1.4 Recognize the definition of and some advantages of computer simulation

A.1.5 Recognize that computers are generally useful for information processing tasks which require any of the following:

a. handling large amounts of information
b. rapid handling of information
c. accuracy
d. repetition
e. the storage of information in an accessible form

A.1.6 Recognize that some of the more important factors which limit and/or restrict computer use include:

a. cost (hardware, software, maintenance, conversion, etc.)
b. people's attitudes
c. unavailability of suitable software/applications
d. hardware limitations--storage limitations, lack of peripherals, etc.
e. complexity of some computer-supported applications

A.1.7 Recognize that innovations in computer hardware and software continually expand the potential utility of the computer.

A.1.8 Identify some of the features of a computer-supported application which make it easy to use.

A.1.9 Recognize some advantages of using a computer for storage and retrieval of information (including continual up-dating and quick recall).

A.1.10 Recognize features and capabilities of personal micro systems as compared to features and capabilities of large data-processing systems.

A.2.1 Determine how computers and computer-supported applications can assist an individual as he or she plays various roles, i.e., consumer, worker, citizen; how such systems assist groups and organizations as they attempt to accomplish tasks and responsibilities.

A.2.2 Assess the feasibility, potential benefits, and hazards of a computer-supported application (cross reference impact).
A.2.3 Design and develop a computer-supported application that would be personally useful.

A.2.4 Recognize some general criteria for computer usage for a particular task (e.g., purpose, time and resource constraints, ethical or moral considerations, relevance).

HARDWARE (H): This domain deals with the basic vocabulary of computer system components.

H.1.1 Recognize that computer hardware refers to the physical components of computer systems.

H.1.2 Recognize that chips are small pieces of silicon that contain electronic logic networks called circuits, and that a computer can be made up of chips mounted on board.

H.1.3 Identify at least five, major functions of a general purpose computer system; namely input, output, storage or memory, control, arithmetic, and timing or clocking.

H.1.4 Recognize that CPU stands for Central Processing Unit, which is the part of the computer which carries out the essential and controlling tasks. The arithmetic, control, and memory units of a computer are generally regarded as comprising the CPU.

H.1.5 Recognize that the term "mainframe" refers to the CPU and its enclosure, or to large computers.

H.1.6 Recognize that a microprocessor is a single chip containing all the electronic logic of a CPU.

H.1.7 Recognize that a microcomputer is a very small computer that uses a microprocessor for its Central Processing Unit.

H.1.8 Recognize that minicomputers are a range of computers larger than microcomputers and smaller than what are usually called "mainframes."

H.1.9 Identify the meaning of acronyms including ROM, RAM, PROM, EPROM, which identify different types of memory.

H.1.10 Recognize the characteristics of secondary storage systems including magnetic tapes, floppy disks, etc.
H.1.11 Recognize that keyboards, punch cards, and tapes serve as input devices.

H.1.12 Recognize that display screens, printers, punch cards, and tapes serve as output devices.

H.1.13 Recognize the function of some basic communications technology (e.g., networks, distributed networks, modem or acoustics-coupler).

H.1.14 Distinguish parallel and serial communications.

H.1.15 Recognize the rapid growth of computer hardware since the 1940s.

H.1.16 Recognize distinctions among the following terms: bit, byte, register, and word.

H.1.17 Recognize both the loose and the precise definition of "K."

H.1.18 Distinguish and be able to identify special-purpose from general-purpose computers.

H.1.19 Identify similarities and differences between programmable calculators and computers.

H.1.20 Distinguish analog and digital computing operations.

IMPACT (I): This domain of computer literacy encompasses the social effects of computerization, including both the positive and negative impacts of computers on society.

I.1.1 Describe some of the more common computer-related careers such as computer programmer, system analyst, computer scientist, computer operator, key operator.

I.1.2 Recognize that computers and computer-supported applications have a major impact on the way ordinary citizens live, work, and play.

I.1.3 Recognize that, for at least the short-term future, computers will continue to be smaller while the amount of information they can hold will continue to increase.

I.1.4 Recognize that computers are used to commit a variety of serious crimes, especially stealing money and information.
1.1.5 Recognize that identification codes (numbers) and passwords are a primary means for restricting use of computer systems, of computer programs, and of data files.

1.1.6 Recognize that the procedures for detecting computer-based crimes are limited.

1.1.7 Identify some advantages or disadvantages of a data base containing personal information on a large number of people; the list might include value for research and potential for privacy invasion.

1.1.8 Recognize several regulatory procedures (e.g., privilege to review one’s own file and restrictions on use of universal personal identifiers), which help to insure the integrity of personal data files.

1.1.9 Recognize that “privacy problems” are characteristic of large information files whether or not they are computerized.

1.1.10 Recognize that computerization both increases and decreases employment.

1.1.11 Recognize that computerization can personalize and impersonalize procedures in fields such as education.

1.1.12 Recognize that computerization can lead to greater independence and dependence upon one’s tools.

1.1.13 Recognize that alleged “computer mistakes” are usually mistakes made by people.

1.2.1 Plan a strategy for tracing and correcting a computer-related error such as a billing error.

1.2.2 Explain how computers make keeping track of people more feasible.

1.2.3 Recognize that even though a person does not go near a computer, he or she is affected indirectly, because society is influenced by computerization.

1.2.4 Explain how computers can be used to impact the distribution and use of economic, social and political power.

1.2.5 Identify and evaluate the positive and negative consequences of computer use in specific
LIMITATIONS (L): This domain focuses on developing a general sense of capabilities and limitations of computing machines. Examples of computer limitations include the fact that computers do not have feelings or consciousness, and are not able to make value judgements.

L.1.1 Recognize that computers are machines, designed, built, and operated by humans to assist in many tasks.

L.1.2 Recognize that computers cannot make value judgement.

L.1.3 Recognize that computers cannot provide answers to every question.

L.1.4 Recognize that despite "artificial intelligence," computers cannot think in the way that we normally use the word "think."

L.1.5 Identify a number of things that computers cannot do or cannot perform efficiently.

L.1.6 Distinguish major differences between human capabilities and computer capabilities.

PROGRAMMING/ALGORITHMS (P): This domain deals with the ability to read, modify, and construct algorithms and programs.

P.1.1 Recognize the definition of algorithm and that flowcharts and programs are alternative forms for expressing algorithms.

P.1.2 Follow and give the correct output for simple algorithms expressed in words, flowcharts, or structure diagrams. The algorithms would include some or all of the following: replacement; calculation; selection (decision points); repetition (simple looping with iteration/recursion); input and output operations.

P.1.3 Given a simple algorithm, explain what it accomplishes (i.e., be able to interpret and to generalize). As in P.1.2., the algorithm could include replacement, calculation, selection, repetition, and input/output.

P.1.4 For simple programs written in a language such as BASIC or PASCAL, follow and give the correct output
for the program. The programs would contain input/output, replacement, calculation, selection, and repetition.

P.2.1 Modify a simple algorithm or program to accomplish a new but related task. This would include changing one or two statements to solve a similar problem, and adding more statements to accommodate additional requirements.

P.2.2 Detect syntax and logic errors in an improperly functioning algorithm or program.

P.2.3 Correct syntax and logic errors in an improperly functioning algorithm or program.

P.2.4 Develop, with appropriate documentation, an algorithm or program for solving a specific problem.

P.2.5 Select an appropriate algorithm from a set of alternatives using criteria such as efficiency, elegance, and appropriateness.

SOFTWARE AND INFORMATION PROCESSING (S): This domain includes vocabulary relevant to software, information processing, and data.

S.1.1 Identify that software refers to computer programs and includes operating systems, compilers, and user (application) programs.

S.1.2 Recognize that digital computers operate upon information which has been encoded in binary, the coding system which can utilize the base 2 number system.

S.1.3 Recognize that a computer needs instructions to operate.

S.1.4 Recognize that a computer gets instructions from a program written in a programming language by a person.

S.1.5 Recognize that computer programs are sets of sequential instructions which perform many tasks such as printing, sorting, calculating.

S.1.6 Identify that the basic elements of an information processing system are input, processing, and output, in that order.
S.1.7 Identify the fact that data are symbols representing things.

S.1.8 Identify the fact that information is data which has been given meaning.

S.1.9 Identify the fact that communication is the transmission of information via coded messages.

S.1.10 Identify the need for data to be organized if it is to be useful.

S.1.11 Identify the fact that data processing involves the transformation of data by means of a set of pre-defined rules.

S.1.12 Recognize that computers process data by searching, sorting, deleting, updating, summarizing, moving.

S.1.13 Recognize that software refers to any nonpermanents sets of programs whereas firmware refers to software that has been made physically permanent.

S.1.14 Recognize that an operating system is software or firmware that performs file managing tasks such as loading programs, copying files, etc.

S.2.1 Select an appropriate attribute for ordering of data for a particular task.

S.2.2 Design an elementary data structure for a given application (that is, provide order for the data).

S.2.3 Design an elementary coding system for a given application.

USAGE (U): This domain involves motor skills for sequencing and execution of tasks on computers or computer terminals.

U.1.1 Connect a microcomputer or computer terminal to a power source and available storage units such as disks or tapes.

U.1.2 Interact with a computer in an online instructional learning situation involving drill and dialogs.

U.1.3 Use system commands for an available computer system (e.g., command to load, list, execute, save, and purge files).

U.1.4 Use program documentation to select and run library programs for specific tasks.
U.1.5 Enter, compile, and debug a simple stored program written in a "higher level" language.

VALUES (V): This affective domain centers on developing positive attitudes toward personal use of computers as well as balanced attitudes toward computers as a social force.

V.1.1 Does not feel fear, anxiety, or intimidation from computer experiences.

V.1.2 Feels confident about his or her ability to use and control computers.

V.1.3 Enjoys and desires work or play with computers, especially computer assisted learning.

V.1.4 Describes past experiences with computers with positive-affect words like fun, exciting, challenging, etc.

V.1.5 Given an opportunity, spends free time using a computer.

V.2.1 Values the potential role of computers in meeting societal and institutional needs (e. g., making full, accurate information available for relevant decisions, problem solutions, or inquiries by private individuals; handling routine data collection/processing/monitoring tasks efficiently).

V.2.2 Values efficient information processing provided that it does not neglect accuracy, the protection of individual rights, or social needs.

V.2.3 Values computerization of routine tasks so long as it frees people to engage in other activities and is not done as an end in itself.

V.2.4 Values increased communication and availability of information made possible through computer use provided that it does not violate personal rights to privacy and accuracy of personal data.
APPENDIX C

DECISION RULES FOR DESIGNATING THE CODING UNITS
DECISION RULES FOR DESIGNATING THE CODING UNITS

The rules for designating the coding units, using the sentence as a unit of analysis, are as follow:

1. Any declarative or interrogative sentence in the main part of the text is designated as a directive statement.

2. Any sentence in segments of an introduction, a review, or a summary part of the text is designated as a miscellaneous statement.

3. Any sentence or combination of words in segments labelled "example" is designated as an example.

4. Any question or exercise in an end-of-chapter segment is designated as a question.

5. Any drawing or picture used to explain or make things clear in the textbook is designated as an illustration.

A count of thought units or propositions (Omanson, Beck, Vass, and McKeown, 1984, p. 49) is particularly applied to an example coding unit for a more meaningful measure. Therefore, the unit quoted hereafter will be counted as three coding units.

Example 3

10 PRINT 10-2+8
20 PRINT 12345
30 PRINT "BASIC" /

when executed, the values of expressions are
displayed on the screen in three consecutive single
space lines as follow:

16
12345
BASIC /

if the list of expressions is omitted, a blank line
is displayed as shown in example 4 /

(p. 27, chapter 4, Textbook B).
APPENDIX D

DETAILED COMPUTATION OF RELIABILITY MEASURE
DETAILED COMPUTATION OF RELIABILITY MEASURE

Without the homogeneity assumption, Cohen (1960) proposed the statistic Kappa, a coefficient of agreement for nominal scale responses between two coders, which is defined as

$$K = \frac{(Po - Pc)}{(1 - Pc)}$$

where $Po$ is the observed proportion of agreement, and $Pc$ is the proportion of agreement expected by chanced. Data collected to assess the agreement between two independent coders can always be represented in a two-dimensional contingency table that categorizes the responses of two coders to the same set of observations. Consider Table 20, suppose that a sample of coding units is rated by two coders, with each unit being assigned to one of the $0$ mutually exclusive categories by two coders. In Table 20 it is assumed that:

1. Coders 1 and 2 both assign $X$ coding units to one of $0$ nominal categories $A_1, A_2, \ldots , A_0$.
2. $X_{ii}$ is the number of coding units placed in category $A_i$ by Coder 1 and Coder 2.
3. $X_{i.}$ is the number of responses in row $i$, $X_{.i}$ is the number of responses in column $i$, and $X_{..} = N$ is the total number of responses given by each coder. In terms of these symbols, the observed proportion of agreement $(Po)$ can be defined as:
Table 20

Illustrative contingency table for agreement measure
between two coders

<table>
<thead>
<tr>
<th>Coder 1</th>
<th>$A_1$</th>
<th>$A_2$</th>
<th>...</th>
<th>$A_0$</th>
<th>Row Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_1$</td>
<td>$X_{11}$</td>
<td>$X_{12}$</td>
<td>...</td>
<td>$X_{10}$</td>
<td>$X_1$</td>
</tr>
<tr>
<td>$A_2$</td>
<td>$X_{21}$</td>
<td>$X_{22}$</td>
<td>...</td>
<td>$X_{20}$</td>
<td>$X_2$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>$A_0$</td>
<td>$X_{01}$</td>
<td>$X_{02}$</td>
<td>...</td>
<td>$X_{00}$</td>
<td>$X_0$</td>
</tr>
</tbody>
</table>

Column Totals: $X_{.,1}$, $X_{.,2}$, ..., $X_{.,0}$, $X_{..}(=N)$

$$P_o = \sum_{i=1}^{c} P_{ii}$$

where

$$\sum_{i=1}^{c} P_{ii} = (X_{11} + \ldots + X_{00}) / X_{..}$$

and the proportion of agreement expected by chance (PC) can be defined as:

$$PC = \sum_{i=1}^{c} P_i P_{.i}$$

where the proportion of all subjects assigned to the $i$th category by Coder 2 is

$$P_{i.} = X_{i.} / X_{..}$$

and the proportion of all subjects assigned to the $i$th
category by Coder 1 is
\[ P_i = X_i / X_{..} \]

To assess the statistical significance of \( K \), the standard error of \( K \) is given by the square root of variance of \( K \), that is

\[ \text{S.E.}(K) = \sqrt{\text{Var}(K)} \]

According to Fleiss's formula, the variance of \( K \) is defined as:

\[ \text{Var}(K) = \left\{ \frac{P_{c} + P_{c}^2 - \sum_{i=1}^{0} P_i P_i (P_i + P_{..})}{N(1 - P_{..})^2} \right\} \]

(Everitt, 1968; Fleiss, Cohen, and Everitt, 1969, cited in Fliess, 1973). The statistical significance of \( K \) is determined by referring to the quantity

\[ Z = K / \text{S.E.}(K) \]

(Fliess, 1973). If \( Z \) is significantly large, the conclusion would be that the observed degree of agreement reflects bona fide reliability. If \( Z \) is small, the conclusion would be that the observed degree of agreement might only reflect random rating.
REFERENCE LIST


Ministry of Education. (1986). Policy for the uses of computers in secondary schools. Speech by Director-General of Ministry of Education in the inauguration of the workshop on computer administration in secondary schools.


