

Development and validation of a
criterion-referenced computer literacy
assessment instrument

by

Mary Frances Montag

A Thesis Submitted to the
Graduate Faculty in Partial Fulfillment of the
Requirements for the Degree of
MASTER OF SCIENCE

Department: Professional Studies in Education
Major: Education (Curriculum and Instructional
Media)

Signatures have been redacted for privacy

Iowa State University
Ames, Iowa

1984

TABLE OF CONTENTS

	<u>Page</u>
CHAPTER ONE: INTRODUCTION	1
Background	1
Statement of the Problem	7
Purpose of the Study	8
CHAPTER TWO: LITERATURE REVIEW	9
Part One: Computer Literacy Definitions and Assessment	9
Part Two: The Development of Criterion-Referenced Tests	22
CHAPTER THREE: METHODOLOGY	56
Step One: Preparation and/or Selection of Objectives	57
Step Two: Preparation of Test Domain Specifications	64
Step Three: Writing Test Items Matched to Objectives	65
Step Four: Preliminary Review of the Test Items	66
Step Five: Determination of Content Validity	66
Step Six: Selection and Additional Editing of Test Items	68
Step Seven: Test Assembly	72
Step Eight: Administration of Final Test	73
Step Nine: Setting Performance Standards	74
Step Ten: Assessment of Reliability and Validity	75

	<u>Page</u>
Step Eleven: Compilation of Test Score Norms	76
Step Twelve: Preparation of User's Technical Manual	77
Summary	77
CHAPTER FOUR: RESULTS	79
Administration of Revised <u>CLAI</u>	79
Collection of Normative Data	80
Performance Standards	83
Reliability	84
Validity	84
Summary	86
CHAPTER FIVE: DISCUSSION OF RESULTS	106
Development of the <u>CLAI</u>	107
Reliability	110
Validity	112
Normative Data	115
Possible Uses of the <u>CLAI</u>	117
Suggestions for Future Research	119
Summary of Results	122
REFERENCES	123
ACKNOWLEDGEMENTS	130
APPENDIX A: SURVEY OF COMPUTER LITERACY COMPETENCIES	132
APPENDIX B: THE COMPLETE UNREVISED LIST OF COMPUTER LITERACY COMPETENCIES	138

	<u>Page</u>
APPENDIX C: DOMAIN SPECIFICATIONS (WITH FIRST SELECTION OF COMPETENCIES)	156
APPENDIX D: JUDGEMENTAL ITEM ANALYSIS FORMS	173
APPENDIX E: REVISED COMPETENCIES KEYED TO TEST ITEMS	182
APPENDIX F: COMPUTER LITERACY ASSESSMENT INSTRUMENT	194
APPENDIX G: NORMATIVE DATA FROM MAURER'S COMPUTER ANXIETY INDEX (1983)	229
APPENDIX H: HUMAN SUBJECTS APPROVAL FORM AND CONSENT FORMS	238

LIST OF TABLES

	<u>Page</u>
Table 1. Means, standard deviations, and ranges of total <u>CLAI</u> and subtests by criterion group	89
Table 2. Percentile scores and corresponding raw scores for <u>CLAI</u> and subtests for instructed criterion group	90
Table 3. Percentile scores and corresponding raw scores for <u>CLAI</u> and subtests for uninstructed criterion group	92
Table 4. Reliability estimates for total <u>CLAI</u> and subtests by criterion group	94
Table 5. Cut-off scores (Berk, 1976) for total <u>CLAI</u> and subtests and percentages of criterion groups classified as masters and nonmasters of the content	95
Table 6. Cut-off scores set by steering committee for total <u>CLAI</u> and subtests and percentages of criterion groups classified as masters and nonmasters of the content	96
Table 7. Comparison of <u>CLAI</u> test scores of instructed versus uninstructed criterion groups	97
Table 8. <u>CLAI</u> test-item analysis data from instructed and uninstructed criterion groups	98
Table 9. Means, standard deviations, and ranges of <u>CAIN</u> scores by norm groups (Maurer, 1983, p.46)	230

LIST OF FIGURES

	<u>Page</u>
Figure 1. Frequency distributions of scores for <u>CLAI</u> Section II, Part One (Beliefs About Computers Scale) by criterion group	101
Figure 2. Frequency distributions of scores for <u>CLAI</u> Section III (Computer Systems) by criterion group	102
Figure 3. Frequency distributions of scores for <u>CLAI</u> Section IV (Computer Applications) by criterion group	103
Figure 4. Frequency distributions of scores for <u>CLAI</u> Section V (Computer Programming) by criterion group	104
Figure 5. Frequency distributions of scores for total <u>CLAI</u> by criterion group	105
Figure 6. <u>CAIN</u> scores of college students (Maurer, 1983, p. 47)	231
Figure 7. <u>CAIN</u> scores of junior high school students (Maurer, 1983, p. 48)	232
Figure 8. <u>CAIN</u> scores of teachers (Maurer, 1983, p. 49)	233
Figure 9. <u>CAIN</u> scores of data processing professionals (Maurer, 1983, p. 50)	234
Figure 10. <u>CAIN</u> scores of users of data processing services (Maurer, 1983, p. 51)	235
Figure 11. <u>CAIN</u> scores of individuals not belonging to any previous group (Maurer, 1983, p. 52)	236
Figure 12. <u>CAIN</u> scores of all groups (Maurer, 1983, p. 53)	237

CHAPTER ONE: INTRODUCTION

Background

In 1978, after perhaps the most complete analysis of the national importance of computer literacy, Andrew Molnar concluded that:

There is a national need to foster computer literacy. Further, if we are to meet this need, we must ensure that high school graduates have an understanding of the uses and applications of the computer in society and its effect upon their everyday lives....A nation concerned with its social needs and economic growth cannot be indifferent to the problems of literacy. If we are to reap the benefits of science-driven industries, we must develop a computer-literate society (Molnar, 1978, p. 14).

Computers are increasingly assuming a pervasive role in society, contributing to such important functions as communications, transportation, education, government, consumerism, entertainment, and employment. This has led to the attitude among many educators that the following are desirable and necessary goals of formal education: an understanding of computers and the impact of computer use and misuse in society; abilities and skills in using computers; and positive anxiety-free attitudes toward the functions of computers in society (Eisele, 1980; Molnar, 1981; Anderson and Klassen, 1981; Masat, 1981).

As early as 1972, the Committee on Computer Education of the Conference Board of the Mathematical Sciences recommended the development of at least one computer literacy course at the secondary level. According to the Committee recommendation, such a course should deal with the ways computers are used, the impact of the uses of computers on individuals and society, the capabilities and limitations of computers, and the concept of the algorithm and its application in flowcharting and programming (Hansen, Klassen, Anderson, and Johnson, 1981).

In recent years, there has been a growing consensus among educators that every student should acquire some computer literacy by the secondary level (Molnar, 1978; Klassen, 1980). The National Association for Educational Computing identified fourteen specific areas of national need in educational computing. One of these needs was, "To develop literacy programs aimed at developing knowledge and appreciation of the use of computers for teachers, students and the general public" (Friedman, 1983, p. 26).

The 1983 report of the National Commission on Excellence in Education, "A Nation at Risk", recommended that a half year of computer science be required of all secondary school students. The report specifically recommended that computer science courses in high school should equip students to: "understand the computer as an

information, computation and communication device; use the computer in the study of the other Basics and for personal and work-related purposes; and understand the world of computers, electronics and related technologies" (National Commission on Excellence in Education, 1983, p. 15).

As part of a Computer Literacy Study conducted by the Minnesota Educational Computing Consortium (MECC), an attempt was made to measure secondary school teachers' opinions of what every student should know about computers. In 1978, a questionnaire about computer use in education was sent to 6,837 secondary mathematics, science, computer science, data processing, and business education teachers in Minnesota. The questionnaire was designed to identify computer users and the types of computer use in secondary schools (Hansen et al., 1981).

Three of the items on the questionnaire were designed to reflect teachers' perceptions of what computer literacy should be. Two of the statements dealt with secondary school students' general knowledge of computers and their role in society. The third item dealt specifically with computer programming. The results indicated that nearly eighty-four percent of the mathematics and science teachers agreed with the statement, "Every secondary student should have some minimal understanding of computers." Nearly ninety-three percent of the teachers agreed with the second

statement, "Every secondary school student should learn about the role that computers play in our society".

The issue of how much programming skill a student should have and the appropriateness of computer programming as a part of computer literacy is probably the most debatable issue in the literature, and specifically, of the MECC study. Forty-three percent of the teachers disagreed with the statement, "Every secondary school student should be able to write a simple program", while just under twenty-nine percent agreed with it. Twenty-five percent of the mathematics teachers and thirty-two percent of the science teachers were undecided on the issue of programming.

The conclusions drawn from the questionnaire included the following: while there was support among mathematics and science teachers for the idea that every secondary school student should have some minimal understanding of computers and the role that computers play in society, these and other general computer literacy topics were not being covered to any great extent in their courses. It was evident from the survey that teachers primarily use the computer as a tool to teach mathematics and science rather than to teach computer literacy concepts.

In response to the need for computer literacy, many educators are struggling to prepare computer literacy

curricula for implementation in the public schools (Bitter, 1982; Hunter, 1980). However, the development of computer literacy curricula is not restricted to the elementary or secondary level. One of the needs identified by the National Association for Educational Computing was, "To develop college preparation programs for computer literacy components that are sensitively designed to prepare computer scientists, teachers for our children today, school administrators, teacher aides and assistants and educational computing specialists" (Friedman, 1983, p. 26). Many higher education institutions have joined the "bandwagon" in an attempt to meet the need for all students to have sufficient computer knowledge, awareness, and skills to function effectively in our technological society (Hunter, 1978; Dennis, 1978).

In addition to the attempt to meet the computer literacy needs of college students, higher education institutions have been challenged to meet the needs of preservice and inservice teachers at all levels and in all subject areas. A nationwide survey of Colleges of Teacher Education (Kull and Archambault, 1984) provided some insights into the nature of the response to the computer education challenge. In 1982, a fifteen item questionnaire was sent to 740 colleges or departments of education which

constitute the national membership of the American Association of Colleges for Teacher Education (AACTE).

The questionnaire addressed the offering of computer courses within teacher preparation curricula. The following conclusions were drawn from the results of the survey:

1. There was a widespread response on the part of colleges and departments of education to the need for providing a computer component as part of the teacher education program.
2. Courses were being offered for both undergraduate and graduate students that included "educational applications" and taught students how to develop their own computer-assisted instruction materials, how to program microcomputers, and also encouraged students to evaluate educational software.
3. Approximately eighty-six percent of the curricular combinations of programming languages included BASIC and forty-seven percent included LOGO.
4. Teachers in all content areas and all levels were enrolled in computer courses. Teachers were typically interested in how they could use computers to enhance their own teaching and provide a more varied learning environment for their students.

A number of institutions reported that computer literacy was or would soon be a general education requirement for all undergraduates. Kull and Archambault (1984) predicted an increase in the number of institutions and states offering certification in computer literacy or requiring competency in computing for general teacher certification. Recently, some universities and colleges

have implemented a computer literacy requirement for all students (Masat, 1981).

The infusion of computer literacy courses at both the secondary and higher education levels presents curriculum developers with the problem of determining the scope of the content appropriate at each level. The wide variety of computer literacy courses being taught at the secondary level has resulted in students entering higher education institutions with a broad range of computer-related experiences and skills. Currently, there are no known standards for determining what aspects of computer literacy should be provided at each level of education, and no effective means of determining what aspects of computer literacy students actually possess at various levels of education.

Statement of the Problem

There is a need to determine the degree of computer literacy possessed by students as they leave secondary schools and enter the higher education institutions. Such an assessment would facilitate the development of computer literacy courses that are appropriate to the needs of the students and would be instrumental in placing the students in courses suited to their level of computer literacy.

Purpose of the Study

The purpose of the research described in this document was to construct a valid and reliable criterion-referenced instrument to assess the computer literacy (knowledge, skills, attitudes) of students prior to and following the taking of computer literacy courses. The Computer Literacy Assessment Instrument (CLAI) was developed to serve three primary purposes:

1. To document the computing competencies of students and to develop normative data about various subgroups of students.
2. To justify revisions in the content of introductory computing courses to reflect student needs and prerequisite skills.
3. To provide criteria for the placement of students in appropriate computing courses based on their level of computer literacy.

CHAPTER TWO: LITERATURE REVIEW

The review of literature is divided into two major sections. The first section deals with research involved with the definition and assessment of computer literacy. The second section deals with research concerning the characteristics and development of criterion-referenced assessment instruments.

Part One: Computer Literacy Definitions and Assessment

This section presents an overview of the computer literacy definitions found in the literature that influenced the operational definition used for the development of the CLAI. The operational definition adopted for this study is discussed in Chapter Three. The second portion of this section reviews the status of computer literacy assessment instruments reported in the literature.

Definitions of computer literacy

One of the most crucial questions facing curriculum developers who are attempting to design computer literacy courses and assessment instruments is: What is "computer literacy"? Prior to the 1970s, the term "computer literacy" was not commonly found in the literature

(Anderson, 1982; Anderson and Klassen, 1981; Hunter, 1980). Recently, the use of the term has become so widespread and varied that serious consideration needs to be given to operationally defining "computer literacy" and to identifying the characteristics of a computer literate individual. The following is an overview of some of the various definitions of computer literacy found in the literature.

The majority of the definitions can be classified according to three overlapping approaches. The proponents of one approach contended that computer literacy was equated with programming skills and the ability to do computing (Luehrmann, 1981). According to this definition, individuals who possess knowledge and awareness of computers and computer applications, but do not have skills in programming or using computers are not considered computer literate.

Proponents of a second approach defined computer literacy as a general knowledge and awareness of what computers are and the use of computers in society. In 1972, the Conference Board of the Mathematical Sciences defined computer literacy as an understanding of computer capabilities, applications, and algorithms (Anderson, 1982). Other educators have included an emphasis on social issues in their definitions of computer literacy. Moursund

(1976) defined computer literacy as the "non technical and low-technical aspects of the capabilities and limitations of computers, and of the social, vocational, and educational implications of computers" (p. 55). The same philosophy is effectively expressed in computer literacy texts for elementary students (Ball and Charp, 1977) and secondary students (Billings and Moursund, 1979).

The third approach combined ideas from both of the other approaches by defining computer literacy as the ability to appropriately use computer applications. The appropriate use of computer applications require a combination of knowledge and awareness of computers as well as the ability to program and do computing.

Watt (1980) defined computer literacy as "that collection of skills, knowledge, understandings, values and relationships that allow a person to function comfortably as a productive citizen of a computer-oriented society" (p. 26). Watt further divided the concept of computer literacy into four distinct, but interrelated, areas:

1. The ability to control and program a computer to achieve a variety of personal, academic and professional goals.
2. The ability to use a variety of preprogrammed computer applications in personal, academic and professional contexts.

3. The ability to understand the growing economic, social and psychological impact of computers on groups within our society and on society as a whole.
4. The ability to make use of ideas from the world of computer applications as part of an individual's collection of strategies for information retrieval, communication and problem solving (p. 26).

Another perspective of the third approach was that computer literacy was a matter of functioning effectively within a given role. In a definition adopted for a computer literacy study, Anderson and Klassen (1981) defined computer literacy as "whatever understanding, skills, and attitudes one needs to function effectively within a given role that directly or indirectly involves computers" (p. 131).

Computer literacy assessment efforts

The research concerned with the assessment of computer literacy and other computer related issues can be divided into three categories according to the type of information measured. Instruments that measure the affective aspects of computer literacy, the cognitive aspects of computer literacy, and both affective and cognitive aspects of computer literacy are reviewed in the following section.

Affective measurements One of the few nationwide computer-related surveys of adults conducted within the last fifteen years was a telephone survey, "A National

Survey of the Public's Attitudes Toward Computers", conducted by AFIPS/Time, Inc. (1971). The results of this survey indicated that in 1971 the public's attitudes were often unrealistic and negative, and the public's understanding of computers was often incorrect. In response to questions about the applications of computers in society, it was found that the public had a general awareness of computer capabilities. However, some people still held misconceptions about computers and the effect computers had on society.

More recently, Ellsworth and Bowman (1982) revised a series of twenty-three questions about computers (seventeen developed by Ahl in 1976 and six added by Lichtman in 1979) to develop a reliable and valid seventeen item Likert-type scale to measure students' beliefs about computers. Ellsworth and Bowman's review of the literature revealed several other scales designed to measure students' attitudes toward computers; however, the scales tended to measure attitudes toward computer-assisted instruction rather than computers in general.

The objectives of the Ellsworth and Bowman (1982) study were to: determine how "computer literate" students would respond to the items (computer literate was operationally defined as junior or senior level computer science majors); determine, using the computer science

students' responses, items having substantial agreement on direction of response; and determine both internal consistency and test-retest reliability of the developed scale using a sample of students who varied in their exposure to computers. The Beliefs About Computers Scale (BACS) was found to have a test/retest reliability of 0.85 and an internal consistency reliability of 0.77 (Ellsworth and Bowman, 1982).

Computer-anxiety is a condition that can influence a person's interaction with computers. A measure of computer anxiety was first developed by Rohner (1981) and then revised and validated by Maurer (1983) at Iowa State University. The resulting Computer Anxiety Index (CAIN) was a twenty-six item Likert-type scale with a test/retest reliability coefficient of 0.90 and an internal consistency reliability estimate of 0.94. Maurer identified two primary potential uses of the CAIN: diagnostic and screening/selection purposes.

Cognitive measurements In a project currently underway at the Human Resources Research Organization (HUMRRO), under a subcontract with the Educational Testing Service, a definition and conceptual structure for computer literacy has been developed and prepared for a document titled "Definition and Conceptual Structure For Computer Literacy" (Hunter, 1983). The purpose of the HUMRRO

project was stated as the development of a pool of questions for the National Center for Educational Statistics to use in instruments designed to survey the nature and extent of computer-related activities and skills among school system superintendents, school principals, elementary and secondary teachers, and elementary and secondary school students during the approximate time frame of 1985-1990 (Hunter, 1983).

The functional definition of computer literacy used in the HUMRRO study was: "Whatever a person needs to know and do with computers in order to function competently in our information-based society" (Hunter, 1983, p. 8). The Computer Literacy Review Panel (CLRP) for the HUMMRO study identified specific tasks that constituted computer literacy and divided these tasks into the following eight domains of computer literacy.

1. Solving problems: Applying and developing algorithms and procedures.
2. Using computer programs to aid in learning school subjects and as a tool for personal work.
3. Writing computer programs.
4. Analyzing computer applications.
5. Understanding social issues.
6. Using computer concepts and terminology.

7. Establishing and administering educational policies.
8. Teaching with and about computers.

The members of the Computer Literacy Review Panel (CLRP) were asked to judge the relative importance of the eight categories of computer literacy for each of the six target groups. Priorities for each target group were established as guidelines for the construction of future instruments. The target groups that were important for the development of the CLAI were primarily students and teachers. For elementary and secondary school students, the CLRP placed the highest priority on the category "Using Computer Programs". "Problem Solving" was the next highest priority, followed by "Writing Computer Programs", and "Understanding Social Issues" (Hunter, 1983). The CLRP placed the highest priorities for teachers on the categories of "Teaching With and About Computers", and "Using Computer Programs". Categories dealing with algorithmic problem solving and understanding social issues were considered to be of secondary importance for teachers.

Affective and cognitive measurements In an extensive review of the literature for the Computer Literacy Study conducted by the Minnesota Educational Computing Consortium (MECC) in 1980, it was reported that previous efforts to construct assessment instruments and

measure computer literacy were limited in scope and were largely unsystematic (Klassen, Anderson, Hansen, and Johnson, 1980). The most systematic effort to assess computer literacy identified by the literature review was conducted by the National Assessment of Educational Progress (NAEP). NAEP's "1977-78 Mathematics Objectives: Second Assessment" included exercises that dealt with several aspects of computer literacy. Both affective and cognitive exercises were included in the assessment that was administered to 13-17 year olds. The results indicated that in 1977 most 13-17 year olds had very little experience with using or programming computers. However, this lack of experience did not necessarily reflect a lack of basic computer knowledge or attitudes about computers (Carpenter, Corbitt, Kepner, Lindquist, and Reys, 1980).

The Computer Literacy Study, conducted by the Minnesota Educational Computing Consortium and funded by the National Science Foundation, resulted in the development of a published set of instructional objectives for computer literacy (Klassen et al., 1980), the construction of the "Minnesota Computer Literacy and Awareness Assessment", and the first statewide testing of computer literacy (Anderson, Krohn, and Sandman, 1980). The research team for the MECC project devoted extensive effort into the development and validation of a

comprehensive, systematic and reliable computer literacy assessment instrument.

The definition of computer literacy proposed by the MECC Computer Literacy Study incorporated knowledge of computers, social implications of that knowledge, and a recognition of the need for skills in communicating with computers. Computer literacy was defined as "whatever understanding, skills, and attitudes one needs to function effectively within a given role that directly or indirectly involves computers" (Anderson and Klassen, 1981, p. 131).

The Computer Literacy Study began with a search of the available literature in an attempt to establish a coherent set of statements that reflected the implications of the phrase computer literacy. Information about a wide range of computer-related courses was collected in the form of course descriptions, course objectives, curriculum guides, and evaluation instruments. In addition, approximately 2000 test items related to computers were collected, judged for quality, and categorized. The collection of computer-related information was then used to develop a comprehensive list of the topics and objectives covered in the courses. The result was a collection of computer literacy objectives that were grouped under six main headings: applications, hardware, impact, programming/algorithms, software and data processing, and

attitudes, values and motivation (Johnson, Anderson, Hansen, and Klassen, 1980). The affective or attitudinal dimension was further divided into such elements as computer enjoyment, computer anxiety, and self-confidence in computing.

The next phase of the Computer Literacy Study was to construct an instrument to assess the computer literacy objectives. While a large number of test items were collected from teacher-made tests, the research team did not find many of those test items to be appropriate for the instrument. The test questions were most often designed to measure student knowledge learned through a specific course or instructional unit. Thus, multiple choice items were written based on the list of Computer Literacy Objectives. The result was a two-part assessment instrument that was initially titled the "Computer Literacy Questionnaire". Part one of the instrument contained items used to form eight attitude scales: enjoyment, anxiety, efficacy, sex-typing, policy concern, educational computer support, social value, and technical value. Part two contained forty-nine test items divided into five cognitive areas: hardware, programming and algorithms, software and data processing, applications, and impact (Klassen et al., 1980).

In 1979, the instrument was administered to a sample of eleventh graders throughout Minnesota as part of the Minnesota State Assessment Program to establish statewide norms (Klassen et al., 1980). The test was shown to be valid and reliable, with an internal consistency (alpha) reliability of 0.89. The field study, in which high school students were tested as they entered and left computer courses, supported the assertion that each of the six dimensions were necessary components of computer literacy (Klassen et al., 1980). The instrument was later re-titled the "Computer Literacy and Awareness Assessment". In a later MECC study involved with the development of computer literacy instructional modules, two more dimensions were added to the original set of six dimensions: limitations and usage. The objectives were also revised for that study and published as the "Computer Literacy Objectives" (Anderson and Klassen, 1981).

Summary

The first part of Chapter Two reviewed a variety of computer literacy definitions found in the literature. The definitions can be categorized into three overlapping approaches: the ability to program and do computing, a knowledge and awareness of computers and what they can do in society, and the ability to use computer applications (requiring both skills in programming or doing computing,

and knowledge and awareness of computers). The review of these definitions contributed to the development of the operational definition of computer literacy for the Computer Literacy Assessment Instrument (CLAI) (see Chapter Three).

The review of the literature also revealed a limited number of studies that attempted to assess computer literacy. Several of the studies were reviewed to determine whether appropriate computer literacy assessment instruments already existed. The computer literacy assessment studies can be divided into three categories: those that assessed affective aspects of computer literacy, those that assessed cognitive aspects of computer literacy, and those that assessed both affective and cognitive aspects of computer literacy. The major studies of importance to the development of the CLAI were the HUMMRO study (Hunter, 1983), the MECC Computer Literacy Study that resulted in the development of the "Minnesota Computer Literacy and Awareness Assessment" (Klassen et al., 1980), Maurer's study (1983) that resulted in the development of the Computer Anxiety Index (CAIN), and Ellsworth and Bowman's study (1982) that resulted in the Beliefs About Computers Scale (BACS). The specific contributions of these studies to the present study are discussed in Chapter Three.

Part Two: The Development of Criterion-Referenced Tests

Introduction

One of the most dramatic advances in the area of educational measurement and evaluation during the past decade has been the shift from norm-referenced to criterion-referenced testing. Although the concept of an absolute versus relative standard of measurement was first introduced through the work of Thorndike in 1913, and further studied by Flanagan, Nedelsky, and Ebel during the 1950s and early 1960s, the term "criterion-referenced measurement" was not used until 1962 when Glaser and Klaus coined the term (Berk, 1980). Glaser (1963) made the following distinction between criterion-referenced and norm-referenced measures:

What I shall call criterion-referenced measures depend upon an absolute standard of quality, while what I term norm-referenced measures depend upon a relative standard (p. 519).

Popham and Husek (1969) further clarified Glaser's distinction between criterion-referenced and norm-referenced measurement and discussed the implications of the two approaches in instructional decision making.

...norm-referenced measures are those which are used to ascertain an individual's performance in relationship to the performance of other individuals on the same measuring device.... Criterion-referenced measures are those which are used to ascertain an individual's status with

respect to some criterion, i.e., performance standard....Criterion-referenced tests are devised to make decisions both about individuals and treatments, e.g., instructional programs (p. 2).

Popham and Husek's 1969 article seemed to stimulate a proliferation of research on the psychometric properties of criterion-referenced tests during the 1970s.

Unfortunately, the resulting body of research exists in published, unpublished, redundant, and fragmented forms. Until recently (Glaser and Nitko, 1971; Harris, Alkin, and Popham, 1974; Millman, 1974; Hambleton and Eignor, 1978), there have been few reliable guidelines for test construction, test assessment, and test score interpretation.

Although standard procedures for norm-referenced testing and measurement are well known to educators, these procedures are inappropriate for criterion-referenced measurements (Glaser, 1963; Hambleton and Novick, 1973; Popham and Husek, 1969; Hambleton and Eignor, 1978). As a result, the available criterion-referenced tests often fall short of the technical quality necessary for them to accomplish their intended purposes. The purpose of this section of Chapter Two is to review research concerned with the development and validation of criterion-referenced tests.

Definition of criterion-referenced measurement

Much of the discrepancy between the magnitude of the efforts to produce criterion-referenced tests and the quality of the tests may be attributed to confusion concerning the definition of criterion-referenced measurement. In 1978, it was reported that there were more than 600 literature citations on the topic of criterion-referenced testing, with almost as many ideas about what a criterion-referenced test was as there were contributors to the field (Hambleton, Swaminathan, Algina, and Coulson, 1978). More than fifty descriptions of a criterion-referenced test have appeared in the research literature since the term was first introduced in 1962 (Berk, 1980). One of the major differences in the various definitions was in the use of the word "criterion". For many people, criterion referred to a performance standard, a minimum proficiency level, or a cut-off score. However, three of the most influential proponents of criterion-referenced testing (Glaser, 1963; Popham and Husek, 1969) used the word criterion to refer to a domain of behaviors. Popham (1975) provided the definition that most accurately reflected a general agreement among researchers about the term (Gray, 1978; Hambleton et al., 1978; Millman, 1974).

A criterion-referenced test is used to ascertain an individual's status (referred to as a domain score) with respect to a well-defined behavior domain (p. 130).

Uses of criterion-referenced tests

Two important uses of criterion-referenced tests are the estimation of examinee domain scores and the allocation of examinees to states such as masters and nonmasters (Hambleton et al., 1978). Masters are defined as examinees who scored above a cut-off score on a test that indicates mastery of the content. Nonmasters are defined as examinees who scored below the cut-off score.

Millman (1974) outlined four major areas of use for criterion-referenced tests:

1. Needs assessment
2. Instruction
3. Program evaluation
4. Teacher improvement and personnel evaluation.

In practical terms, criterion-referenced tests are being used to monitor individual progress through objectives-based instructional programs, to diagnose learning deficiencies, to evaluate educational and social action programs, and to assess competence on certification and licensing examinations.

Steps in the development of criterion-referenced tests

A twelve step model for the development and validation of criterion-referenced tests was first presented by Hambleton and Eignor in 1979 and was later recommended by Hambleton (1980). The model was very similar to Fremer's (1974) outline for the construction of criterion-referenced tests (Hambleton et al., 1978) and has been a generally accepted procedure for the development of criterion-referenced tests.

Steps for Developing Criterion-Referenced Tests and Validating Test Score Uses (Hambleton, 1980, p. 83)

1. Preparation and/or selection of objectives or domain specifications.
2. Preparation of test specifications (for example, available time, selection of objectives to be measured by the test, number of test items/domain specification, appropriate vocabulary, method of scoring).
3. Writing test items "matched" to objectives.
4. Preliminary review of test items.
5. Determination of content validity of the test items.
 - a. Involvement of content specialists.
 - b. Collection and analysis of examinee response data (field test-item analysis).
6. Additional editing of test items.
7. Test assembly.
 - a. Determination of number of test items/domain specification.
 - b. Test item selection.
 - c. Preparation of directions and sample questions.
 - d. Layout and test booklet preparation.
 - e. Preparation of scoring keys.
 - f. Preparation of answer sheets.
8. Setting standards for interpreting examinee test performance.
9. Test administrations.

10. Assessment of test score reliability and validity; compilation of test score norms (optional).
11. Preparation of a user's manual and a technical manual.
12. Periodic collection of additional technical information.

Review of criterion-referenced measurement research

Domain specifications and item generation The first and most important step in the development of a criterion-referenced test is to operationally define the domain of content or behaviors the test is intended to measure. This step is crucial to the next step, that of generating test items representative of the domain of content. The characteristics of the objectives or domain specifications contribute to the type and quality of test items generated. In addition, the validity and interpretation of the test scores are contingent upon the precision of the domain specifications. Thus, it is important that the relationship of the scores to the domain be systematically considered during the design of the test.

In a recent state of the art report, Berk (1980) emphasized the contention of leading proponents of criterion-referenced measurement that the traditional test development approach of defining a body of content that includes a content outline, a list of objectives, and a table of specifications tended to produce an ambiguous domain specification for criterion-referenced tests.

During the past decade, domain specification strategies have been devised to overcome the deficiencies of the traditional approach by attempting to provide an unambiguous definition of a domain and implicitly or explicitly constitute sets of rules for generating the items. Theoretically, well-written domain specifications will simultaneously constrain item writers so that they produce congruent items and communicate effectively to test interpreters (Popham, 1980).

In the same report, Popham (1980) reviewed four specification strategies for criterion-referenced tests. Unfortunately, research dealing with the particulars of test specifications was almost nonexistent. Lacking a solid research base, Popham was forced to rely heavily on his personal experiences with projects carried out by the Instructional Objectives Exchange (IOX), an agency specializing in the development of criterion-referenced tests. The following is a brief explanation of the four specification strategies discussed by Popham (1980).

Behavioral objectives The original intent of IOX in 1968 was to serve as a clearinghouse for the generation of behavioral objectives to guide educators in the development of tests. Popham identified five IOX criteria for selecting behavioral objectives to guide the

development of a test (in Klein and Kosecoff 1973, p. 16).

1. Transferability Within Domain. The form of learner behavior selected should be the most generalizable of those represented in the content general domain, i.e., a learner mastering the designated behavior requirements would likely be able to transfer that mastery to most, if not all, of the other eligible behavioral requirements in the content general domain.
2. Widely Accepted. The objectives selected should be the most widely accepted as important by those in the field. Unlike the IOX objective collections where we present a wide array of alternatives and then encourage educators to choose among them, here we will have to go with the majority preference.
3. Terminality. If there is a degree of possible hierarchy present in the contending types of behaviors under consideration, such that some are considered precursive or enroute to others, the chosen specific objective should represent the most terminal learner behavior.
4. Transferability Outside the Domain. Another consideration in selecting a specific objective is the degree to which that behavior, once mastered, will be transferable outside the content general domain, for example, to domains which might be learned by students in the future.
5. Ease of Scorability. In an effort to produce tests which have considerable practical utility, we must try to select learner behaviors which, other factors being equal, can be easily scored by those educators employing them.

Regardless of the efforts of IOX researchers to use behavioral objectives as domain specifications for the

development of test items, the IOX staff soon realized that behavioral objectives were inadequate as constraining devices for test item writers. Behavioral objectives left too many decisions to the test writer. This resulted in the generation of incongruent items for the same objective (Popham, 1980).

Amplified objectives Millman (1974) defined an amplified objective as "an expanded statement of an educational goal that provides boundary specifications regarding testing situations, response alternatives and criteria of correctness". The advantage of the additional guidelines added to a behavioral objective was that they helped to define the relevant domain of test items. However, Popham contended that although the IOX amplified objectives did constrain the test item writers considerably, the item writers and test interpreters still had too much latitude to allow them to write congruent items (Popham, 1980).

Item forms An item form is a highly detailed set of rules for generating test items of a homogeneous nature. An item form has the following characteristics. It generates items with a fixed syntactical structure; it contains one or more variable elements; and it defines a class of item sentences by specifying the replacement sets for the variable elements (Hambleton et al., 1978; Popham,

1980). An item form precisely defines a sampling domain of hypothetically equivalent test items.

Hively, Patterson, and Page (1968) used an item form scheme in a generalizability study of a "universe-defined" system of arithmetic achievement tests. The major contribution of the Hively et al. study (1968) was that it demonstrated that it was possible to develop and use item generation rules to construct a criterion-referenced test. However, the study also emphasized one of the major weaknesses of item generation strategies; that is, the strategies are more easily employed with highly structured subject matter areas such as mathematics.

The IOX item writers found item forms to be very detailed and constraining. However, the efforts to employ the item form scheme resulted in the generation of too many item forms for the behaviors they were attempting to measure and too few item writers willing to use the "too technical" item forms (Popham, 1980).

IOX test specifications The IOX staff continued to search for a method of writing domain specifications for criterion-referenced tests that would retain the descriptive rigor of item forms, yet be more practically implemented by item writers and teachers. A limited focus measurement strategy (Popham 1978) was adopted to create a limited number of test descriptors that

were sufficiently specific to communicate properly to test users and sufficiently targeted to be of utility to teachers.

Popham (1980) identified five components of a set of IOX test specifications: general description, sample item, stimulus attributes, response attributes, and specification supplement. The general description was a one or two sentence statement of what the test was intended to measure. The purpose of the general description was to provide a succinct overview of the set of behaviors in the domain. This was sometimes referred to as an objective. The sample item was included to provide item format cues for the item writers. Stimulus attributes identified the influential factors that constrained the composition of a set of test items. Response attributes focused on the examinee's response to the elements generated according to the stimulus attributes section. The specification supplement section included any set of content information that was necessary for the item writers to generate the appropriate items.

Comparisons of domain specification strategies

Berk (1979) rated seven domain specification strategies (behavioral objectives, amplified objectives, item forms, IOX test specifications, item transformations, algorithms,

and mapping sentences) according to eight practicality factors: clarity, simplicity, availability, development time, development cost, adaptability, domain appropriateness, and practicality. Berk concluded that the seven strategies varied considerably in the extent to which they achieved an unambiguous domain definition and overcame the deficiencies of objectives. Amplified objectives, IOX domain specifications, and mapping sentences were characterized as achieving less ambiguous definitions and indicated the most potential for practical use by teachers and evaluators; while item transformations, item forms, and algorithms attained unambiguous definitions by means of sophisticated rule structures and were judged to be the most impractical to use.

Logsdon (1981) conducted a study that examined the effectiveness of two different domain specification strategies in achieving homogeneity of criterion-referenced test items. The study examined the extent to which test writers following the IOX domain specification strategy for a cognitive skill generated items that were more homogeneous than those generated from the simple objective strategy for the same skill.

The results indicated that the IOX domain specification strategy generated more homogeneous items across writers than the abbreviated strategy. A

homogeneous set of items indicated a more clearly defined behavior domain. However, the writers who followed the IOX domain specifications found the specifications highly explicit, laborious to develop, and tedious to follow when generating items (Logsdon, 1981).

Validation of criterion-referenced tests In general, the quality or validity of criterion-referenced test items can be determined by the extent to which they reflect the domain from which they were derived. Specific validity considerations are important during three stages of criterion-referenced measurement: the selection of objectives or domain specifications, the measurement of objectives included in the criterion-referenced test, and the interpretation and uses of the test scores.

A content validity analysis should be completed at the test development stage to determine the match between the item content and the objectives the items are intended to measure. The content validity of a test will positively influence the construct (decision) validity of the test. Construct validation studies are important in assessing the validity of test scores for accomplishing their intended uses. Specifically, decision validity is important if the test scores will be used to make decisions such as assigning examinees to "mastery states" (Hambleton, 1980).

The content validity of a criterion-referenced test is generally established through two item analysis approaches: a judgmental item analysis of test items by content specialists who establish item-objective congruence, and an empirical item analysis designed to express the degree of relationship between the intent of each item and the responses of students to each item (Hambleton et al., 1978; Hambleton, 1980; Berk, 1980). The purposes of an item analysis are to identify structural flaws, determine whether each item measures its respective objectives (item-objective congruence), and to determine whether the items function consistently with their intended purposes, that is, whether the items differentiate between masters and nonmasters of those objectives (Berk, 1980).

Judgmental item analysis Hambleton (1980)

believed that the method of establishing the content validity of criterion-referenced test items that held the most merit involved the judgments of content specialists regarding the degree of item-objective congruence for each test item. In this method, content specialists were asked to judge the items in relation to this question: "Is the format and content of an item appropriate to measure some part of the domain specification?" (Hambleton, 1980).

A study by Ryan (1968) examined the question, "Can the content specialists make meaningful judgments about the

relevance of items to instructional content?" Ryan concluded that teachers can make judgments about test items on two dimensions: the relevance of the items to the instructional content, and the difficulty of the item.

Three techniques for the collection and analysis of content specialists' judgments were described in Rovinelli and Hambleton's 1976 study designed to achieve two purposes:

1. To generate and organize appropriate judgmental data techniques and methods of data analysis and reporting.
2. To examine three different techniques for the collection of judgmental information with regard to the type, reliability, and validity of the information provided (Rovinelli and Hambleton, 1976).

The first method suggested by Rovinelli and Hambleton (1976) involved asking content specialists to rate test items relative to each objective from the domain specification. A rating technique developed by Hemphill and Westie (1950) for use in constructing personality tests was employed. With this method, three ratings were possible to indicate the degree of feeling a rater had that an item was a measure of an objective: -1 indicated a definite feeling that an item was not a measure of the objective, 0 indicated uncertainty, and +1 indicated a strong feeling that an item was a measure of the objective. A statistic called the Index of Item-Objective Congruence

was derived from the Index of Homogeneity of Items developed by Hemphill and Westie (1950). The Index of Item-Objective Congruence has been shown to be a valid procedure for collecting and analyzing judgmental data on item validity. The value of the Index could serve as a cutting score for judging the item-objective match of each of the test items.

The second procedure employed the use of a rating scale. With this method, content specialists used a rating scale to rate the appropriateness or suitability of each item as a measure of the intended objective or domain specification. For example, a Likert type scale of 1 to 5 could be used, with 1 indicating low congruence, and 5 indicating high congruence between the item and the objective. The mean and median rating across content specialists was an indication of the degree of congruence between the item and the intended objective.

A matching task was the third procedure introduced by Rovinelli and Hambleton (1976). With this method, content specialists were presented with two lists, one with test items and the other with objectives or domain specifications. Each content specialist was asked to indicate which objective each test item measured. A contingency table was then developed by calculating the number of content specialists matching each item to each

objective. A chi-square test for independence was used to analyze the frequency of each item-objective match. The item-objective pairs with the highest frequency of matches were considered the best items.

All of these techniques were shown to provide information that could be used to ascertain if an item were a measure of an objective. However, there were differences in the types of data that were collected through the use of these techniques. Rovinelli and Hambleton (1976) recommended the use of the Hemphill-Westie procedure over the other two techniques because the Index of Item-Objective Congruence provided a meaningful interpretation of the extent to which an item is judged to be a valid measure of the intended objective.

However, there are some disadvantages to the use of the Hemphill-Westie procedure that may be rectified through the use of another technique. For instance, the Hemphill-Westie procedure cannot be used to collect information about the quality of the item or the characteristics of distractors. The procedure is also quite time consuming if a large number of objectives and items are used (Rovinelli and Hambleton, 1976).

The judgment of content specialists can also be used to determine the representativeness of the test items to the domain of content measured by the test. For the test

to be content valid, the items selected for the final version of the test must be representative of the entire domain. However, this judgment can not be determined until the final item selection decisions have been made and the test is completed (Hambleton, 1980).

Empirical item analysis The second approach for determining content validity, empirical item analysis, involves the calculation of various item statistics such as item difficulty, item discrimination, and item homogeneity obtained from a pilot test of the items with criterion groups (Hambleton et al., 1978; Berk, 1980). The steps of a pilot test include the selection of criterion groups, the administration of the test items, the collection of informal student feedback, and the computation of item statistics mentioned above (Berk, 1980).

The groups of subjects selected for a pilot test are often referred to as criterion groups because their selection is based on some criterion of current or future performance. Two categories of criterion groups are typically of interest to criterion-referenced test developers: students who would be successful on the items (masters) and students who would be unsuccessful on the items (nonmasters). In a comprehensive review of item statistics in 1978, Berk found that the majority of the statistics were based on pretest-posttest repeated

measurements from one group of students who were considered nonmasters prior to instruction and masters following instruction, or two independent measurements from two different criterion groups, uninstructed nonmasters and instructed masters.

The difficulty of an item is determined by the percentage of students who answered the item correctly. The difficulty index can range in value from 0 to 100, with the higher the index, the easier the item. In general, the item difficulty index values for criterion-referenced test items should indicate that the items are relatively difficult for an uninstructed criterion group and easy for an instructed criterion group. In other words, the difficulty index should range from 0-50 for the uninstructed group and 70-100 for the instructed group (Berk, 1980).

The index of item discrimination measures the difference in performance between pretest and posttest criterion groups or uninstructed and instructed criterion groups. It is assumed that uninstructed students are more likely to answer an item incorrectly, while instructed students are more likely to answer the same item correctly. A criterion-referenced test should maximize the discriminations between criterion groups and minimize

discriminations among individuals within any one group (Glaser, 1963).

The numerous discrimination indices described in the literature suggest that there was no single best method for estimating discrimination. Berk (1980) extracted four approaches that were conceptually and computationally simple yet statistically sound and evaluated them according to practicality and meaningfulness. The four methods evaluated were the following:

1. Pretest-posttest difference (DISppd) - the proportion of students who answered the item correctly on the posttest minus the proportion of students who answered the item correctly on the pretest.
2. Uninstructed-instructed group difference (DISuigd) - the proportion of students in the instructed group who answered the item correctly minus the proportion of students in the uninstructed group who answered the item correctly.
3. Individual gain (DISig) - the proportion of students who answered the item incorrectly on the pretest and correctly on the posttest.
4. Net gain (DISng) - extends the individual gain by considering performance of all students who answered the item incorrectly on pretest. It is the individual gain minus the proportion of students who answered incorrectly on both occasions.

All four of these indices yield a value ranging from -1.00 to +1.00. The pretest-posttest difference and the uninstructed-instructed group difference were found to be

the simplest to determine because they were computed directly from the difficulty levels for the criterion groups. The individual gain and the net gain were found to be the most sensitive to pretest-posttest item gain scores. The selection of which index to use may depend on the criterion groups available for the analysis (Berk, 1980).

An empirical item analysis can provide construct validity information as well as content validity information. Construct validity can be demonstrated by the amount of difference between the performance of students who have been instructed and the performance of students who have not been instructed in the domain of content measured by the instrument. The larger the difference between the two groups, the more construct validity the test contains (Hambleton, 1980).

When criterion-referenced test scores are used to describe examinees or make decisions regarding the assignment of examinees to mastery groups, it is essential to establish the validity of the descriptions and decisions. This type of validity is known as construct decision validity. Decision validity requires the setting of a standard of performance and the comparison of test performance of two or more criterion groups in relation to the specified standard. Decision validity is determined by summing the percentage of instructed examinees who exceed

the performance standard (classified as masters) and the percentage of uninstructed examinees who do not exceed the performance standard (classified as nonmasters). The higher this sum, the higher the number of students correctly classified as masters and nonmasters, and the more decision validity the test contains (Hambleton, 1980).

Test developers need to consider the information desired and the resources available before selecting the type of judgmental or empirical procedure to use to determine the content and construct validity of a criterion-referenced test. Regardless of the method employed to determine content and construct validity, it is crucial that the appropriate revisions are made when discrepancies between an objective and its respective item are identified.

Item selection Once items have been judged logically by content specialists and empirically tested through item analysis, a decision is made to either accept the item for the final test item sample, revise the item before inclusion in the final test item sample, or eliminate the item from the final test item sample. The guidelines for selecting criterion-referenced items for a final test item sample can be summarized in the following way (Berk, 1980):

<u>Item characteristic</u>	<u>Criterion</u>	<u>Index value</u>
Item-objective congruence	Matches objective being assessed	none
Difficulty	Difficult for uninstructed group	0-50%
	Easy for instructed group	70-100%
Discrimination	Discriminates between instructed and uninstructed criterion groups	high positive (depends on index)

If an item does not meet one or more of the above criteria, a decision must be made to retain it, revise it, or discard it. Some important considerations of this decision are discussed in the following section (Berk, 1980).

1. An item that is not congruent with its objective(s) should not be included in the test regardless of all other characteristics.
2. If an item is congruent with its objective but the item statistics indicate that it fails to discriminate between the criterion groups, the item should be retained when one or both of the following conditions exist: due to the specificity of the behavioral objective, no other item could be written to measure the objective, the low or zero discrimination index suggests that the item may validly measure the absence of an instructional effect.
3. Selecting only the best discriminating (high positive) items would produce the best test in terms of decision validity, but the content validity may be compromised.
4. It is inappropriate to evaluate item effectiveness purely on statistical grounds; a more comprehensive interpretation of

indices that take into account the objective being measured, the students being tested, and the instructional program is recommended.

5. The difficulty and discrimination indices of items selected should reflect a logical trend; that is, the difficulty indices should be lower for uninstructed students and there should be a positive discrimination index. Items that meet these criteria should be identified first. All other items require further analysis.
6. If the difficulty index for an item is higher or lower than expected, possible explanations include faulty objectives or faulty instruction. The objective or instruction may need to be revised or discarded rather than the item.
7. A negative discrimination index indicates a faulty item that may need revision. The item may be ambiguous, contain more than one correct answer, or contain ineffective distractors.
8. A negative discriminating item that possesses no visible technical flaws should be discarded.

If the judgmental item analysis and/or the empirical item analysis reveal that a multiple choice item is faulty, the items should be further analyzed to determine how it should be revised. Items yielding low positive, zero, or negative discrimination indices require an inspection of the examinees' responses to each of the item's distractors. The criteria for evaluating a choice response pattern are the following (Berk, 1980):

1. Each distractor should be selected by more examinees in the uninstructed group than in the instructed group.

2. At least a few uninstructed examinees should choose each distractor.
3. No distractor should receive as many responses by the instructed group as the correct answer.

After test items have been identified for the sample of test items, item selection for particular tests is then a matter of randomly selecting items from the sample. The random selection of items from a well-defined domain of items makes it possible for "strong" criterion-referenced interpretations of the test scores (Millman, 1974).

Standard setting methods A performance standard or cut-off score is a point on a test score scale that is used to categorize examinees into two groups (masters and nonmasters) that reflect different levels of proficiency relative to a particular objective or set of objectives measured by the test. It is important that care and attention be given to the setting of standards for even the most technically sound and valid test (Hambleton, 1980).

Millman (1974) reviewed methods of setting passing scores or standards by categorizing the methods into five sources of information; the performance of others, item content, educational consequences, psychological and financial costs, and errors due to guessing and item sampling. A similar review was presented by Hambleton (1980), who further categorized the methods into

judgmental, empirical, and combinational. Hambleton (1980) and Millman (1974) both agreed that all standard setting methods involved judgment and were arbitrary, making the standard setting process exceedingly complex and subjective.

Judgmental methods Standard setting methods involving the performance of others depend on the criterion that a predetermined percent of the examinees pass or are assigned to the mastery group. These methods are most applicable when the number of individuals who can or should be given a treatment or a certification is fixed. The performance of individuals who are already identified as being certified as masters can be used to establish the standard score for others. By administering the test to the predetermined mastery group, a standard score can be set to correspond to some percentile on the score distribution of the mastery group (Millman, 1974).

There are a number of standard setting methods that consider the item content in making judgments about the items that masters should be able to answer correctly. Each test item is inspected by judges and a judgment is made concerning how important it is that each item be answered correctly by minimally competent individuals.

In 1954, Nedelsky proposed a method that required judges to identify distractors in multiple-choice items

that minimally competent students should have been able to eliminate as incorrect. The reciprocal of the remaining alternatives was calculated as the minimum passing level (MPL) for each item (Millman, 1974; Hambleton, 1980; Poggio, Glassnap, and Eros, 1981).

Another judgmental method concerned with item content was developed by Ebel in 1979. In this method, judges rated items along two dimensions, relevance (essential, important, acceptable, and questionable) and difficulty (easy, medium, and hard). These levels formed a 3 x 4 grid in which each item was assigned. The judges were asked to make a third judgement by assigning a percentage to each cell to indicate the percentage of items that the minimally competent examinee should have been able to answer. Finally, the number of test items in each cell was multiplied by the appropriate percentage. The standard score for each judge was determined by the sum of all of the cells divided by the total number of test items. The composite standard score was the average of all standards for all judges (Millman, 1974; Hambleton, 1980; Poggio et al., 1981).

A third judgmental method, developed by Angoff in 1971, asked judges to assign a probability (0-100) to each test item indicating the probability that minimally competent examinees would answer the item correctly. These

probabilities were then converted to proportions and summed. The average sum across judges was the standard score (Millman, 1974; Hambleton, 1980; Poggio et al., 1981).

Combination methods The Borderline-Group and Contrasting-Group methods presented by Zieky and Livingston in 1977 were procedurally similar in that the judgments were about students rather than items (Hambleton, 1980). Zieky and Livingston contended that judging individuals was likely to be a more familiar task for teachers than judging items.

The Borderline-Group method required that judges first defined the minimally acceptable performance on the content area being assessed. The test was then administered to a group of examinees who were identified as being on the borderline between masters and nonmasters of the content. The median test score for this group was identified as the standard (Hambleton 1980).

The Contrasting-Groups method was similar to the Borderline-Group method, except it required the judges to identify those students they were certain were either definite masters or nonmasters of the skills measured by the test. The test was administered to both groups and the test score distributions for the two groups were plotted.

The standard was identified as the point of intersection of the two distribution lines (Hambleton, 1980).

Berk proposed a method in 1976 that was very similar to the Contrasting-Groups method. It involved the consideration of test responses of instructed and uninstructed groups of students. Berk contended that the optimal cutting point for a criterion-referenced test could be located by identifying the intersection of the test score distribution lines for the instructed and uninstructed groups (Hambleton, 1980).

An empirical study of the Nedelsky, Ebel, Angoff, and Contrasting-Groups methods of setting standards was conducted by Poggio, Glasnapp, and Eros (1981). The results demonstrated that the use of a single method to set a performance standard was arbitrary and that none of the methods studied were superior to any other method.

Reliability of criterion-referenced tests

"Reliability refers to the extent to which measurement results are free of unpredictable kinds of error" (Morris and Fitz-Gibbon, 1978). Estimates of reliability answer the question: Does the instrument yield consistent results? A demonstration of reliability is necessary but not conclusive evidence that an instrument is valid.

Methods for demonstrating that an instrument is reliable involve the comparison of one administration of

the instrument with another administration to the same group of people. A statistic, usually a correlation coefficient, is calculated to demonstrate the degree of similarity of the two sets of results. The higher the correlation, the smaller the influence of error and the more reliable the instrument is. Reliability is expressed as a reliability coefficient, a number between 0.00 and 1.00.

Several methods for demonstrating reliability are available. The characteristics of the instrument and the availability of resources such as money and time usually influence the decision as to which method to use to demonstrate reliability. The most common methods are discussed briefly in the following section.

Test-retest reliability Test-retest

reliability refers to the ability of the test to yield consistent results across multiple administrations of the test. The test-retest method of demonstrating reliability involves the administration of the instrument to the same group of people twice. The second administration must occur within a time period during which the ability, attitude, or skill measured by the instrument is not expected to change. It is usually recommended that the second administration occur one month after the first. A second administration of the instrument that occurs within

a few days of the first administration causes a problem because some of what the students remember from the first administration may carry over to the second administration. A longer interval between the two administrations increases the possibility that the skill or attitude of the respondents will have changed. Other methods of demonstrating reliability have been developed to alleviate the influence of memory effects or real changes (Morris and Fitz-Gibbon, 1978).

Alternate-form reliability The alternate-form method of demonstrating reliability is similar to the test-retest method, except it alleviates the problem of memory effects by using two equivalent forms of the same instrument. Each form of the instrument is administered to each respondent on two different occasions. This method does not completely eliminate the effects of memory because the format of the test across the two forms remains the same. The alternate form method also requires that time be spent writing extra items and preparing two test forms (Morris and Fitz-Gibbon, 1978).

Internal consistency reliability Internal consistency refers to the tendency of different items to elicit the same ability or attitude from any respondent on a single administration of the instrument. The split-half method of demonstrating reliability yields a measure of

test consistency within a single administration. The instrument is divided into two halves and administered to one group of students. Each half is treated as a separate administration. This method separates reliability considerations from the effects of learning or developmental change. The split-half method is best used with instruments that have many items, and where pairs of items can be considered equivalent enough for random distribution to essentially separate forms of the test. A high split-half reliability indicates that the test is internally consistent (Morris and Fitz-Gibbon, 1978).

Two sets of formulas are widely used to estimate internal consistency reliability of a single test from a single administration: the Spearman-Brown Formulas, and the Kuder-Richardson Formulas (Ebel, 1972). The Spearman-Brown Formulas are used to predict the reliability of a lengthened test, assuming that the material added to the test is highly similar to that already present in it. One of the most common uses of the Spearman-Brown Formulas is in obtaining an estimate of the reliability of an entire test by dividing the test into two halves and treating one half as the original test and the other half as the lengthened test. The formula requires the reliability coefficient of the original test, the reliability

coefficient of the lengthened test, and the number of times that the original length of the test has been increased.

The Kuder-Richardson Formulas yield an estimate of internal consistency with a single administration of an instrument, without actually splitting the test into two halves. The Kuder-Richardson Formulas require the following information: the number of items in the test, the standard deviation of the test scores, and the difficulty of each item in the test or the average difficulty of all items as reflected in the mean test score. The Kuder-Richardson formulas are widely used in the estimation of test reliability because of their convenience and their statistical soundness (Ebel, 1972).

Iowa State University's Test and Evaluation Services standardly uses the Kuder-Richardson 20 formula to estimate test reliability. The following suggestions are offered to prospective test writers by the Testing and Evaluation Service. In general, more test items and more test items with higher discrimination indices improve the Kuder-Richardson 20 reliability coefficient for a test. Norm-referenced tests that are the sole factor in the evaluation of students should have a reliability of 0.85 or higher. Norm-referenced tests that will be one of several factors used to evaluate a student should have reliabilities of no lower than 0.70. Tests that have

qualities of both criterion-referenced and norm-referenced measurement should be expected to yield lower Kuder-Richardson 20 reliability coefficients than those of strictly norm-referenced tests, because reliability is a function of variance. Tests that have qualities of criterion-referenced measurement typically have reduced variability in the scores (Iowa State University Test and Evaluation Services, 1983).

Summary

This part of Chapter Two reviewed research concerned with the development of criterion-referenced tests. A twelve step model was presented as a guide for the development of the criterion-referenced Computer Literacy Assessment Instrument (CLAI). Research was reviewed to identify the various options within each step that were considered when designing an appropriate methodology for this study. The specific methodology used in this study is discussed in Chapter Three.

CHAPTER THREE: METHODOLOGY

The procedures for the construction of a valid and reliable instrument to assess the computer literacy of undergraduate students are discussed in this chapter. The following steps, proposed by Hambleton and Eignor in 1979 (Hambleton, 1980), were used in the development of the Computer Literacy Assessment Instrument (CLAI).

Steps for Developing Criterion-Referenced Tests and Validating Test Score Uses (p. 83)

1. Preparation and/or selection of objectives.
2. Preparation of test domain specifications (for example, available time, selection of objectives to be measured by the test, number of test items/domain specification, appropriate vocabulary, method of scoring).
3. Writing test items "matched" to objectives.
4. Preliminary review of test items.
5. Determination of content validity of the test items.
 - a. Involvement of content specialists.
 - b. Collection and analysis of examinee response data (pilot test-item analysis).
6. Additional editing of test items.
7. Test assembly.
 - a. Determination of number of test items/domain specification.
 - b. Test item selection.
 - c. Preparation of directions and sample questions.
 - d. Layout and test booklet preparation.
 - e. Preparation of scoring keys.
 - f. Preparation of answer sheets.
8. Setting standards for interpreting examinee test performance.
9. Test administrations.
10. Assessment of test score reliability and validity; compilation of test score norms (optional).

11. Preparation of a user's manual and a technical manual.
12. Periodic collection of additional technical information.

Step One: Preparation and/or Selection of Objectives

The first step in the development of the criterion-referenced CLAI was to define the domain of content to be measured by the instrument so that appropriate objectives or competencies could be identified. A review of the literature was conducted to determine a definition of computer literacy, identify computer literacy competencies, and identify any existing test items or instruments measuring computer literacy.

Operational definition of computer literacy

The review of the literature and consultations with a steering committee for the development of the CLAI (consisting of several instructional computing experts at Iowa State University) resulted in the adoption of the following definition of computer literacy.

Computer literacy is operationally defined as 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. The knowledge and skills of computer literacy as defined above have been divided into four major categories; computer attitudes, computer systems, computer applications, and computer

programming. The four sections are defined below.

1. Computer Attitudes refer to an individual's feelings about the personal and societal use of computers in appropriate ways. Positive attitudes include an anxiety free willingness or desire to use the computer, confidence in one's abilities to use the computer, and computer responsibility.
2. Computer Systems refer to the appropriate, knowledgeable use of equipment (hardware) and programs (software) necessary for computer applications. This requires understanding and abilities in the following areas: computer functions, computer hardware, computer software, computer systems configuration, computer terminology, historical development, and the operation of computers.
3. Computer Applications refer to the ability to responsibly evaluate, select, and implement a variety of practical computer applications to do meaningful and efficient work based on an understanding of the following: general types of applications, capabilities and limitations of applications, societal impact (past, present, and future), evaluation and selection techniques, and specific applications (word processing, data base management, spreadsheet/financial management, statistical analysis, graphics, and educational applications).
4. Computer Programming refers to the ability to direct the operation of the computer through the skillful use of programming languages (high-level as well as software languages). This requires an understanding of problem solving strategies, algorithms and flowcharts, languages, and programming skills.

Computer literacy competencies

A collection of computer literacy competencies reported in the literature were considered as possible competencies for the proposed instrument. The competencies collected were derived from a wide variety of sources, such as computer literacy course outlines, textbooks, curriculum guides for public school districts and higher education institutions, and computer literacy research studies. "The Computer Literacy Objectives", developed as part of the Computer Literacy Study conducted by the Minnesota Educational Computing Consortium (Klassen et al., 1980), were reviewed by the steering committee and accepted for the initial list of computer literacy competencies. Other competencies that were recommended frequently in the literature and that were consistent with the definition were also selected.

In addition to the list of competencies derived from the literature, an attempt was made to ensure a more comprehensive and valid list of competencies for the domain specification. A survey was sent to 327 computer education specialists. Each content specialist was asked to read the purpose of the proposed instrument, the operational definition of computer literacy, and then write two computer literacy competencies for each of the four categories of the definition (see Appendix A for a sample

of the survey letter). Surveys were sent to individuals selected from the following sources:

1. The participants of the National Computer Literacy Goals for 1985 Conference (Seidel, Anderson, and Hunter, 1982).
2. The presenters at the National Educational Computing Conference of 1982 (Smith and Moum, 1982).
3. Computer Consultants listed in the 1983 Classroom Computer News Directory of Educational Computing Resources, Part III: Local and Regional Resources (Kelman, 1983).
4. The members of the International Council for Computers in Education Organization listed in the April 1983 issue of The Computing Teacher (Moursund, 1983).
5. Names of instructors derived from a listing of colleges and universities offering summer school computer courses in the March 1982 issue of The Computing Teacher (Moursund, 1982).

Approximately 90 useable surveys were returned. No follow up letters were sent to the nonresponders because of the restriction of a limited budget. The competencies identified by the respondents of the surveys were collected, sorted and categorized into a comprehensive list. This list was revised and edited several times to eliminate repetitions of similar competencies.

The list of competencies obtained from the computer specialists was then combined with the list of competencies identified through the literature and a tally of each discreet competency was made (see Appendix B). The entire

list of competencies was then reviewed by the CLAI steering committee. The competencies were selected based on the consideration of the IOX Criteria for the Selection of Objectives (Klein and Kosecoff, 1973).

1. Transferability Within the Domain (appropriateness of the competency to the definition of computer literacy was considered).
2. Widely Accepted (by content specialists and in the literature).
3. Terminality (consideration was given to Bloom's levels of Behavioral Objectives, so that high cognitive levels were included in the competencies).
4. Transferability Outside the Domain.
5. Ease of Scorability (competencies that could be measured by multiple choice items were selected over competencies that could not be measured by multiple choice items).

A total of eighty-seven competencies were selected for inclusion in the domain specifications. The steering committee judged the competencies selected to be representative of the domain of content defined as computer literacy. The competencies were categorized into the four subsections of the definition (nine for Computer Attitudes, twenty-four for Computer Systems, thirty-three for Computer Applications, and twenty-one for Computer Programming). The complete list of computer literacy competencies

selected are included with the Domain Specifications in Appendix C.

Identification of existing instruments

Three valid and reliable computer-related instruments were identified for consideration of possible inclusion in the CLAI. "The Minnesota Computer Literacy and Awareness Assessment" (Klassen et al., 1980), The Computer Anxiety Index (CAIN) (Maurer, 1983), and the Beliefs About Computers Scale (BACS) (Ellsworth and Bowman, 1982) were reviewed by the steering committee to determine if any portions of the instruments were consistent with the computer literacy definition and competencies for the CLAI.

It was the opinion of the steering committee that "The Minnesota Computer Literacy and Awareness Assessment" was not appropriate for this project. Most of the items on the instrument were knowledge or comprehension level questions. One of the goals identified by the CLAI steering committee was to develop an instrument that included items that measured high level cognitive skills as identified by Bloom's Taxonomy of Educational Objectives (Bloom, Englehart, Furst, Hill, and Krathwohl, 1956). Bloom and his associates identified six levels of educational objectives in the cognitive domain: knowledge, comprehension, application, analysis, synthesis, and evaluation. An attempt was made to develop an instrument

that measured high level computer literacy skills, such as application, analysis, synthesis, and evaluation skills, in addition to lower level skills.

Although the Minnesota Computer Literacy Study that resulted in the development of "The Minnesota Computer Literacy and Awareness Assessment" defined computer literacy to include skills in the use of computers and computer applications, the instrument did not contain items that attempted to measure such skills. The instrument seemed to measure only the knowledge and awareness aspects of MECC's definition of computer literacy. The CLAI steering committee concluded that higher level items could be written to measure computer-related skills as well as computer-related knowledge and awareness.

The CAIN (Maurer, 1983) and the BACS (Ellsworth and Bowman, 1982) were also reviewed by the steering committee and were considered to be appropriate for the Computer Attitudes section of the CLAI. Both of these instruments consisted of Likert-type items and were shown to be valid and reliable measures of students' attitudes and anxiety towards computers. The CAIN is a twenty-six item scale with a test/retest reliability of 0.90 and an internal consistency reliability estimate of 0.94 (Maurer, 1983). The BACS is a seventeen item scale with a test/retest

reliability of 0.85 and an internal consistency reliability of 0.77 (Ellsworth and Bowman, 1982).

Step Two: Preparation of Test Domain Specifications

Test domain specifications for the CLAI were developed based on the suggestions proposed by Popham (1980) and others (Berk, 1980; Millman, 1974; Hively et al., 1968). A format similar to the Amplified Objective Method suggested by Millman (1974) and the IOX Test Specification Method proposed by Popham (1980) was used because of the complexity of the content domain of computer literacy. The length of time required to prepare other more specific domain specifications was prohibitive.

The domain specifications for the CLAI (see Appendix C) contained a description of the purpose of the instrument, the definition of computer literacy, the list of competencies, guidelines for writing effective multiple choice items, and sample items. The test domain specifications also contained information pertaining to the length of the test, the number of items desired, and the method of scoring the test.

Step Three: Writing Test Items Matched To Objectives

The test domain specifications were distributed to a team of fifteen computer literacy content specialists in the department of Professional Studies in Education at Iowa State University. Each content specialist was asked to write multiple choice questions for a subset of the total number of competencies in the Computer Systems, Computer Applications, and Computer Programming sections (ten or eleven items for each writer). Two questions were written for each competency in the three sections mentioned above. This produced a total of 186 questions. No new items were written for the Computer Attitudes section because the CAIN (Maurer, 1983) and the BACS (Ellsworth and Bowman, 1982) were used for the Computer Attitudes section of the instrument. The Computer Attitudes section was divided into two parts, Part One (the BACS) and Part Two (the CAIN), each yielding an individual score.

The final instrument was designed to be computer scored, so the items were all multiple choice or Likert-type. Writers were encouraged to incorporate diagrams, illustrations, and other creative devices into appropriate questions to ensure a variety of test items at all levels of Bloom's Taxonomy of Educational Objectives (Bloom et al., 1956).

Step Four: Preliminary Review of the Test Items

The 186 test items written by the content specialists were reviewed, edited and revised by the principal researcher. The entire collection of items was divided into two sets. Each set of items contained questions related to the competencies (ninety-one questions in set A and eighty-five questions in set B) plus fourteen items asking for demographic information about the students.

Step Five: Determination of Content Validity

Empirical item analysis

A pilot test was conducted to collect data for an item analysis. The two sets of test items were each administered to two criterion groups as recommended by Berk (1980): an instructed group of students (masters) and an uninstructed group of students (nonmasters).

Subjects The instructed (master) group consisted of forty-four subjects: forty-one students enrolled in Secondary Education 102 and Secondary Education 302 and three staff members of the College of Education. The students in these courses were considered instructed in computer literacy because Secondary Education 101 (Introduction to Computer Applications) is recommended as a

prerequisite to each course. Set A was administered to twenty subjects and Set B was administered to twenty-four subjects from the instructed criterion group.

The uninstructed (nonmaster) group consisted of thirty-five Secondary Education 301 and Curriculum and Instructional Media 505 students who had not taken a computer literacy course. Students in these classes who reported that they had taken a computer course were not included in the data analysis. Set A was administered to twenty-one subjects and Set B was administered to fourteen subjects from the uninstructed criterion group.

The data from each set of items, each subsection of the test, and each criterion group were analyzed by obtaining the following statistics: descriptive statistics (means, variances, standard deviations), item difficulty indices, the uninstructed-instructed group difference discrimination indices (DISuigd), the correlations between the score on each item and the total score, and the frequencies of students responding to each distractor on each item.

Judgmental item analysis A judgmental item analysis was conducted by a panel of nine computer specialists at Iowa State University. The content specialists were asked to examine each item and its respective competencies to determine the item-competency

congruence and the technical quality of the items. A rating scale like one proposed by Rovinelli and Hambleton (1976) was used to rate how well each item measured the competency it was intended to measure (1-poor, 2-fair, 3-average, 4-good, 5-excellent).

A three point rating scale was used to rate the technical quality of each item. The technical rating indicated whether the content specialist thought the item should be rejected (1), accepted with revisions (2), or accepted as it was (3). The judgmental item analysis forms that were used are in Appendix D. The mean ratings for all of the judges were calculated as an indication of the degree of item-competency congruence for each item. The higher the number, the more congruent the item-competency match.

Step Six: Selection and Additional Editing of Test Items

Test questions were accepted for inclusion in the final test, revised for inclusion in the final test, or eliminated from the final test based on the item analysis and the judgments of the content specialists. Berk's (1980) guidelines for the selection of items for a criterion-referenced instrument were considered in the decision to retain, revise, or reject each item.

<u>Item characteristic</u>	<u>Criterion</u>	<u>Index value</u>
Item-competency congruence	Matches objective being assessed	mean rating (3.00-5.00)
Difficulty (percent correct)	Difficult for uninstructed group	0-50%
	Easy for instructed group	50-100%
Discrimination (DISuigd)	Discriminates between instructed and uninstructed criterion groups	positive index (+10-+100)
Discrimination (item-score correlation)	Discriminates within each criterion group	positive index (+.10-+1.00)

1. An item that is not congruent with its objective(s) should not be included in the test regardless of all other characteristics.
2. If an item is congruent with its objective but the item statistics indicate that it fails to discriminate between the criterion groups, the item should be retained when one or both of the following conditions exist: due to the specificity of the behavioral objective, no other item could be written to measure the objective, the low or zero discrimination index suggests that the item may validly measure the absense of an instructional effect.
3. Selecting only the best discriminating (high positive) items would produce the best test in terms of decision validity, but the content validity may be compromised.
4. It is inappropriate to evaluate item effectiveness purely on statistical grounds; a more comprehensive interpretation of indices that take into account the objective being measured, the students being tested, and the instructional program is recommended.

5. The difficulty and discrimination indices of items selected should reflect a logical trend; that is, the difficulty indices should be lower for uninstructed students and there should be a positive discrimination index. Items that meet these criteria should be identified first. All other items require further analysis.
6. If the difficulty index for an item is higher or lower than expected, possible explanations include faulty objectives or inadequate instruction. The objective or instruction may need to be revised or discarded rather than the item.
7. A negative discrimination index indicates a faulty item that may need revision. The item may be ambiguous, contain more than one correct answer, or contain ineffective distractors.
8. A negative discriminating item that possesses no visible flaws should be discarded.

If the judgmental item analysis and/or the empirical item analysis revealed that a multiple choice item was faulty, the item was further analyzed to determine whether it should be discarded or revised. Items that yielded low positive, zero, or negative discrimination indices required an inspection of the examinees' responses to each of the item's distractors. The criteria used for evaluating a choice response pattern were (Berk, 1980):

1. Each distractor should be selected by more examinees in the uninstructed group than in the instructed group.
2. At least a few uninstructed examinees should choose each distractor.

3. No distractor should receive as many responses by the instructed group as the correct answer.

As a result of the judgmental and empirical item analysis, a total of eighty items were selected for the three subsections of the test: twenty-nine for computer systems, twenty-eight for computer applications, and twenty-three for computer programming. To ensure that the items included in the final instrument were representative of the domain of content of computer literacy, items were selected so that each competency identified in the domain specification was measured by the test. The competencies had already been judged to be representative of the domain by the CLAI steering committee. In order to establish a representative sample of test questions measuring all of the competencies, many items were extensively revised and some additional items were written.

The test items for each subsection of the instrument were sequenced from those measuring low level cognitive skills to those measuring high level cognitive skills in order to facilitate the interpretation of student scores. In an attempt to order the test questions within each subsection, it became apparent that the competencies should be ordered in the same manner to facilitate relating the items to their respective competencies. In the process of sequencing the competencies, a number of the competencies

were revised to more accurately reflect the actual competencies measured by the test. Competencies that expressed the same basic skills were combined into one competency, and competencies that expressed more than one skill were separated into discrete competencies. The final list of Computer Literacy Competencies contained eighty competencies: nine in Computer Attitudes, twenty-five in Computer Systems, twenty-five in Computer Applications, and twenty-one in Computer Programming (see Appendix E).

Step Seven: Test Assembly

The final version of the Computer Literacy Assessment Instrument (CLAI) was prepared (see Appendix F). The CLAI was divided into five sections: Background Information, Computer Attitudes, Computer Systems, Computer Applications, and Computer Programming. The Computer Attitudes section was further divided into two parts; Part One was the Beliefs About Computers Scale (Ellsworth and Bowman, 1982), and Part Two was the Computer Anxiety Index (Maurer, 1983). The instrument was designed so that it can be administered as a whole, or by subtest. The following steps were included in the preparation of the test booklet:

1. Preparation of layout of test booklet.
2. Preparation of final test (typing, printing).

3. Preparation of scoring keys.
4. Preparation of answer sheets.

Step Eight: Administration of Final Test

The CLAI (except Part Two of the Computer Attitudes section) was administered to two groups of subjects, instructed (masters) and uninstructed (nonmasters), to demonstrate the test's validity and reliability, and to determine normative data. Part Two of the Computer Attitudes section, the CAIN (Maurer, 1983), was omitted from this administration because validity, reliability, and normative data had been collected by Maurer in 1983. It was not considered necessary to repeat what had been accomplished previously. The normative information for the CAIN is included in Appendix G.

Subjects

The instructed group consisted of 152 Secondary Education 101 students who had received instruction based on the computer literacy competencies and were considered masters of computer literacy. The uninstructed group consisted of 110 Secondary Education 301 students who had not taken a computer course based on the computer literacy competencies and were considered nonmasters of computer literacy.

Step Nine: Setting Performance Standards

All known methods for setting performance standards are judgmental and arbitrary (Hambleton, 1980; Millman, 1974); and no one method has been found to be superior over another (Poggio et al., 1981). The performance standards, or cut-off scores, for the total CLAI and its three cognitive subtests were determined following the administration of the test to the two criterion groups (uninstructed and instructed) by the method proposed by Berk (1976). This method, similar to the Contrasting-Groups Method (Hambleton, 1980), involved the identification of the point of intersection of the frequency distribution curves for the uninstructed and instructed groups.

The test score corresponding to the point of intersection for each section of the test was then evaluated by the principal researcher and the steering committee to determine whether it was a reasonable and appropriate criterion for classifying students as masters and nonmasters of computer literacy. If the performance standard identified with this method was not considered appropriate, a standard was set based on the judgment of the steering committee. Research designed to set more appropriate standards may be necessary in the future.

Step Ten: Assessment of Reliability and Validity

Reliability

Internal consistency reliability coefficients for the total instrument, the three cognitive subtests, and each criterion group were estimated using the Kuder-Richardson 20 reliability formula (Ebel, 1972; Iowa State University Test and Evaluation Services, 1983). Internal consistency for Section II, Part One (BACS) was estimated with coefficient alpha (Cronbach, 1970). Since the instrument has characteristics of both norm-referenced and criterion-referenced tests, a reliability of greater than 0.70 was considered an acceptable reliability coefficient (Iowa State University Test and Evaluation Services, 1983).

Validity

Construct validity for the total instrument and each of its subtests was determined by calculating the amount of difference between the mean scores of the instructed and the uninstructed groups. The means for the instructed group should be significantly higher than the means for the uninstructed group. A statistically significant ($p < 0.05$) difference between the two means was considered necessary to demonstrate that the difference in the means did not occur by chance. A T-test was used to determine if the

differences between the means of the two groups were significant at the 0.50 level (Mason and Bramble, 1978).

Decision validity for the entire instrument and each subtest of the instrument was determined by summing the percentage of uninstructed students who were classified as nonmasters of the content and the percentage of instructed students who were classified as masters of the content. Students who scored above the cut-off score were classified as masters of the content, and students who scored below the cut-off score were classified as nonmasters of the content. The number of instructed students classified as masters and the number of uninstructed students classified as nonmasters should be high to demonstrate that the decisions made based on the test scores are valid decisions. In other words, the higher the total percentage of students who performed as was expected, the more decision validity the test contained (Hambleton, 1980).

Step Eleven: Compilation of Test Score Norms

Normative data for the two criterion groups, uninstructed and instructed college students, were collected for the entire instrument and each subtest. The mean score, standard deviation, range, frequency

distribution, and percentile scores were reported for the entire test and its subtests for each criterion group.

Step Twelve: Preparation of User's Technical Manual

A User's Technical Manual was prepared as part of the complete CLAI Package. The User's Technical Manual included the following sections:

1. Statement of the purpose of the instrument.
2. The definition of computer literacy.
3. The Computer Literacy Competencies keyed to the test items.
4. Reliability information.
5. Validity information.
6. Normative data.
7. The Computer Literacy Assessment Instrument.
8. Answer keys.

Summary

A twelve step model for the development of a criterion-referenced test (Hambleton, 1980) was used as a guide for the development of the CLAI. An attempt was made to construct as valid and reliable an instrument as the available time and resources would allow. All of the steps

of the methodology were carefully designed so as to contribute positively to the overall validity and reliability of the completed instrument.

CHAPTER FOUR: RESULTS

The procedures for the development and validation of a reliable Computer Literacy Assessment Instrument (CLAI) were completed successfully as described in the previous chapter. The steps included the identification of computer literacy competencies, the writing of test items matched to the competencies, the pilot testing of the test items, and the administration of the revised instrument to criterion groups to establish reliability, validity, performance standards, and normative data.

Administration of Revised CLAI

The revised CLAI was administered to two criterion groups, students instructed in computer literacy and students uninstructed in computer literacy. Section II, Part Two, the CAIN (Maurer, 1983), was not given to these subjects because validity, reliability, and normative information had been collected from a similar group of students by Maurer in 1983. The normative information from Maurer's 1983 study are reported in Appendix G.

Collection of Normative Data

The means, standard deviations, and ranges for the CLAI and its subtests for each criterion group are reported in Table 1. The mean scores for the instructed group were higher than the mean scores for the uninstructed group by one to two standard deviations for all sections of the test except Section II, the BACS (Ellsworth and Bowman, 1982). The mean scores for the BACS suggested that both instructed and uninstructed subjects had more positive attitudes about computers than negative attitudes since lower scores indicated positive attitudes toward computers. On a scale of 1-6 with 1 being the most positive, the mean scores converted to average ratings were 2.27 for the instructed group and 2.50 for the uninstructed group.

The percentile scores and corresponding raw scores for the total test and its subtests for each criterion group are presented in Tables 2 and 3. These tables can be used to compare students' scores on subsequent testings of the CLAI with similar or differing norm groups. The change in percentile scores for similar norm groups over many subsequent testings could be used to substantiate that students in a particular norm group, as a whole, are becoming more or less computer literate. The percentile scores of students could be used as a guide for placing

students in instructional treatments based on whether they score above or below a specified percentile.

The frequency distributions of raw scores for the total CLAI and each subtest for both criterion groups are shown in Figures 1 through 5. The differences in the frequency distributions between the two criterion groups can be more easily seen and evaluated if the two distributions are shown on the same graph. The points of intersection of the two frequency distribution curves for each section of the test were identified as the cut-off scores for classifying students as masters and nonmasters of the content measured by each section of the instrument.

Item statistics, difficulty and discrimination indices, for each CLAI test item were collected for both criterion groups (uninstructed and instructed) and are reported in Table 8. The Difficulty Indices were determined by computing the percentage of students in each group who answered the item correctly. A range of 0 to 100 was possible, with 100 indicating the lowest level of difficulty.

Two different Discrimination Indices were computed for each item. The first was the correlation between the item score and the total test score. A range of -1.00 to +1.00 was possible with +1.00 indicating the highest level of discrimination between the students within each group. In

other words, an item with a high discrimination index indicated that the students who scored high on the total test tended to answer the item correctly, and the students who scored low on the total test tended to answer the item incorrectly. The second Discrimination Index was the Uninstructed-Instructed Group Difference Discrimination Index (DISuigd) (Berk, 1980). It was computed by subtracting the Difficulty Index for the uninstructed group from the Difficulty Index for the instructed group, yielding an index of -100 to +100. The higher the index, the more the item discriminated between students in the instructed group and students in the uninstructed group. In other words, an item with a high positive DISuigd index indicated that the instructed students tended to answer the item correctly, and the uninstructed students tended to answer the item incorrectly.

The Difficulty and Discrimination Indices for each item need to be examined carefully to identify items that might be improved by further revisions of the instrument in the future. The same criteria should be used to determine faulty items as were used in the item analysis conducted during the pilot test (Chapter Three).

Also included in Table 8 are the competencies measured by each item, the item-competency congruence rating from the judgmental analysis, and the number of subjects who

omitted each item. An item-competency congruence rating was not reported for items that were written following the judgmental analysis. However, they were written based on the suggestions of the content specialists so were considered to be congruent with the competencies.

Performance Standards

Performance standards, or cut-off scores, for each cognitive subtest of the CLAI, as well as, the total test were determined using Berk's (1976) method for setting performance standards discussed in Chapter Three. The scores corresponding to the point of intersection between the frequency distribution curves for the two criterion groups were found for each of the three cognitive subtests. The resulting cut-off scores are reported in Table 5.

The steering committee evaluated the cut-off scores established with this method and concluded that the cut-off scores were not appropriate cut-off scores for classifying students as masters and nonmasters of computer literacy. The steering committee felt that computer literate individuals should be able to answer correctly more of the items than the cut-off scores suggested, thus the cut-off scores should be higher to adequately identify computer literate individuals. The mean score for the instructed

group was recommended as a more appropriate cut-off score for each section of the test. The cut-off scores recommended by the steering committee are reported in Table 6.

Reliability

Internal consistency reliability estimates were calculated for the total test and each of its subtests for the two criterion groups. The reliability coefficients for the three cognitive subtests and the total test were calculated using the Kuder-Richardson 20 reliability formula (Ebel, 1972). The reliability coefficients for Section II, Part One, (BACS) (Ellsworth and Bowman, 1982), were calculated using Cronbach's formula for coefficient alpha (Cronbach, 1970). The reliability coefficients are reported in Table 4.

Validity

Content validity for the test items was established during the development of the instrument. The use of computer specialists to identify competencies representative of the domain of content, the writing of test items matched to the competencies, and the item and

judgmental analysis all contributed positively to the content validity of the test.

Construct validity was demonstrated for the total test and its subtests by calculating the differences between the mean scores of the instructed and uninstructed groups and showing that the differences were statistically significant. The actual differences between the two criterion groups revealed that the mean of the instructed group was at least one standard deviation above the mean of the uninstructed group for every section except Section II, the BACS. T-tests were calculated to test five post-hoc null hypotheses:

1. There is no significant difference between the CLAI mean scores for the instructed and uninstructed criterion groups.
2. There is no significant difference between the CLAI Section II mean scores for the instructed and uninstructed criterion groups.
3. There is no significant difference between the CLAI Section III mean score for the instructed and uninstructed criterion groups.
4. There is no significant difference between the CLAI Section IV mean score for the instructed and uninstructed criterion group.
5. There is no significant difference between the CLAI Section V mean score for the instructed and uninstructed criterion group.

All of the null hypotheses were rejected at the .001 level of significance (Table 7). Thus, there were significant differences between the mean scores of the two

criterion groups for the CLAI and its subtests. This indicates that the differences did not occur by chance and helps to establish that the instrument is construct valid.

The procedure for establishing decision validity proposed in Chapter Three was followed using both the cut-off scores set with Berk's (1976) method and the cut-off scores recommended by the steering committee. Decision validity, the validity of the decisions made on the basis of student scores, is considered high when a high percentage of the students scored as their criterion groups would indicate they should. In other words, the higher the percentage of instructed students who scored above the cut-off score and the higher the percentage of uninstructed students who scored below the cut-off score, the higher the decision validity of the instrument. The cut-off scores and corresponding percentages are reported in Tables 5 and 6. Both cut-off scores yielded percentages of appropriately classified subjects that were considered to be high enough to indicate a high degree of decision validity.

Summary

The procedures followed to develop a valid and reliable instrument to measure computer literacy (as

defined in Chapter Three) were very successful. The content validity of the instrument was established during the development phase through the use of content specialists who identified computer literacy competencies, wrote test items, and rated how well the items measured the competencies. A statistical item analysis helped to identify items in need of revision or deletion. As a result, the instrument was considered to have a high degree of content validity. In other words, the test actually measured what it was supposed to measure (i.e., computer literacy competencies).

The test was shown to have a high degree of construct validity. Students who were given instruction based on the computer literacy competencies and were considered to possess a high degree of computer literacy had a significantly higher mean score on the test than students who were not instructed in computer literacy and were considered not computer literate. Construct validity is supposed to indicate that the test scores are valid and can be used for their intended purposes. The CLAI test scores can validly be used to identify computer literate and noncomputer literate individuals.

Decision validity is a specific type of construct validity that indicates that the decisions made on the basis of the test scores are valid. The CLAI was shown to

have a high degree of decision validity. A high percentage of the subjects were appropriately classified as masters or nonmasters of computer literacy.

The CLAI was found to have a high degree of internal consistency reliability. The reliability coefficients for the total test were 0.86 for the instructed group and 0.91 for the uninstructed group. The reliability coefficients of the subtests were lower, but still indicated internal consistency reliability.

Table 1. Means, standard deviations, and ranges of total CLAI and subtests by criterion group

	Section II ^a Attitudes	Section III Systems	Section IV Applications	Section V Programming	Total ^b test
Mean score					
Instructed ^c	38.70	18.62	17.50	11.09	47.20
Uninstructed ^d	42.50	11.70	10.82	5.72	28.24
Standard deviation					
Instructed	9.49	3.83	4.31	3.79	10.18
Uninstructed	8.19	5.11	5.10	4.59	13.15
Actual range					
Instructed	25-64	8-27	0-26	0-20	13-69
Uninstructed	23-62	0-25	0-23	0-19	2-63
Possible range	17-102	0-29	0-28	0-23	0-80

^aPart One, the Beliefs About Computers Scale (Ellsworth and Bowman, 1982): the lower the score, the more positive the attitude toward computers.

^bTotal score for Sections III, IV, and V combined.

^cN=152.

^dN=110.

Table 2. Percentile scores and corresponding raw scores for CLAI and subtests for instructed criterion group

%tile	Raw Scores by Section					Total ^e test
	II ^a		III ^b	IV ^c	V ^d	
	Part 1	Part 2				
100	64	-	27	26	20	69
99	63	5.0	26	25	19	68
98	59	-	-	-	18	-
97	56	4.6	25	-	-	64
95	53	-	24	24	-	62
94	-	4.0	-	-	17	-
93	-	-	-	23	-	61
92	-	3.8	-	-	-	-
91	49	3.6	23	-	16	-
90	-	-	-	-	-	60
89	-	-	-	22	-	-
88	48	-	-	-	-	-
87	-	-	-	-	-	59
86	-	3.4	-	-	15	-
85	-	-	-	-	-	58
84	47	-	22	21	-	-
82	-	-	-	-	-	57
81	46	-	-	-	14	56
78	45	3.2	-	-	-	55
77	-	-	-	20	-	-
76	-	-	-	-	-	54
75	44	-	21	-	-	-

^aComputer Attitudes. Part 1 is the Beliefs About Computers Scale (Ellsworth and Bowman, 1982) with a possible range of 17-102, 17=most positive computer attitude. Part 2 is the Computer Anxiety Index (Maurer, 1983) with a possible range of 1-6, 6=highest level of computer anxiety.

^bComputer Systems (maximum possible score=29).

^cComputer Applications (maximum possible score=28).

^dComputer Programming (maximum possible score=23).

^eSections III, IV, V combined (maximum possible score=80).

Table 2. (continued)

%tile	Raw Scores by Section					Total test
	Part 1	II Part 2	III	IV	V	
72	43	-	-	-	13	53
70	42	-	-	-	-	-
69	-	3.0	-	-	-	-
68	41	-	-	-	-	-
66	-	-	-	-	-	52
64	-	-	-	19	-	-
63	-	-	20	-	12	51
61	40	-	-	-	-	-
59	-	2.8	-	-	-	50
58	-	-	19	-	-	-
57	-	-	-	-	11	-
56	39	-	-	-	-	49
55	-	-	-	18	-	-
53	-	-	-	-	-	48
50	-	-	-	-	-	47
49	38	-	18	17	-	-
47	-	2.6	-	-	-	-
46	-	-	-	-	10	46
43	-	-	-	-	-	45
41	37	-	-	-	-	-
40	-	-	17	16	-	-
38	36	-	-	-	-	44
35	-	2.4	-	-	-	43
34	35	-	-	-	9	-
31	-	-	-	-	-	42
30	34	-	-	-	-	-
28	-	-	16	15	8	41
24	33	2.2	-	-	-	40
22	-	-	15	-	-	38
14	31	2.0	14	13	-	-
12	30	-	-	12	-	34
9	29	-	-	-	-	33
8	-	1.8	-	-	-	-
4	26	1.6	-	-	-	29
3	25	-	11	9	-	27
2	5	-	10	-	4	-
1	2	1.4	9	7	3	26

Table 3. Percentile scores and corresponding raw scores for CLAI and subtests for uninstructed criterion group

Percentile	Raw Scores by Section					Total ^e test
	II ^a		III ^b	IV ^c	V ^d	
	Part 1	Part 2				
100	62	-	25	23	19	63
99	61	5.0	24	-	18	-
98	-	-	-	-	17	62
97	58	4.6	23	22	-	58
96	57	-	21	-	16	57
95	56	-	20	21	15	55
94	-	4.0	-	-	13	-
93	-	-	19	19	-	52
92	54	3.8	-	18	12	-
91	53	3.6	-	-	-	48
90	52	-	-	-	-	45
89	-	-	18	17	-	-
88	-	-	17	-	-	44
86	-	3.4	-	-	11	43
85	-	-	-	16	10	42
83	50	-	-	-	-	39
81	-	-	-	-	9	38
80	49	-	16	-	-	37
79	-	-	-	14	-	36
78	48	3.2	-	-	-	-
77	-	-	15	-	-	35
75	-	-	-	13	8	-

^aComputer Attitudes. Part 1 is the Beliefs About Computers Scale (Ellsworth and Bowman, 1982) with a possible range of 17-102, 17=most positive computer attitude. Part 2 is the Computer Anxiety Index (Maurer, 1983) with a possible range of 1-6, 6=highest level of computer anxiety.

^bComputer Systems (maximum possible score=29).

^cComputer Applications (maximum possible score=28).

^dComputer Programming (maximum possible score=23).

^eSections III, IV, and V combined (maximum possible score=80).

Table 3. (continued)

File	Raw Scores by Section					Total test
	II Part 1	II Part 2	III	IV	V	
74	-	-	-	-	-	34
69	-	3.0	-	-	7	-
68	46	-	13	-	-	33
63	-	-	-	-	-	31
60	-	-	-	-	-	30
59	44	2.8	-	11	-	29
56	-	-	-	-	5	-
51	43	-	-	-	-	27
50	-	-	-	10	-	-
47	-	2.6	11	-	-	-
46	42	-	-	-	-	-
45	-	-	-	9	4	25
43	-	-	10	-	-	-
42	41	-	-	-	-	24
39	40	-	-	-	-	23
36	-	-	-	-	3	-
35	39	2.4	9	8	-	-
31	38	-	-	-	-	-
28	-	-	-	-	2	20
26	36	-	8	-	-	-
25	-	-	-	7	-	19
24	-	2.2	-	-	-	-
23	-	-	-	-	1	-
20	-	-	7	-	-	18
15	-	-	6	6	-	17
14	33	2.0	-	-	-	-
13	-	-	-	5	0	14
10	31	-	5	-	-	12
8	-	1.8	-	4	-	10
7	-	-	4	-	-	9
5	28	-	3	2	-	8
4	-	1.6	2	-	-	5
3	27	-	-	1	-	3
2	25	-	1	0	-	2
1	23	1.4	0	-	-	-

Table 4. Reliability estimates for total CLAI and subtests by criterion group

Criterion group	Section II ^a Attitudes	Section III Systems	Section IV Applications	Section V Programming	Total ^b test
Instructed	0.81 ^c	0.64 ^d	0.75	0.69	0.86
Uninstructed	0.73	0.78	0.80	0.81	0.91

^aPart One, the Beliefs About Computers Scale (Ellsworth and Bowman, 1982).

^bSections III, IV, and V combined.

^cCoefficient alpha.

^dKuder-Richardson 20 reliability coefficient (Sections III, IV, V, and total).

Table 5. Cut-off scores (Berk, 1976) for total CLAI and subtests and percentages of criterion groups classified as masters and nonmasters of the content

Section of test	Percentage of criterion groups appropriately classified as masters and nonmasters of the content			
	Cut-off ^a score	Uninstructed ^b (N=110)	Instructed ^c (N=152)	Both groups ^d (N=262)
III. Systems	14	68	89	81
IV. Applications	13	70	88	81
V. Programming	6	56	95	79
Total test	37	78	84	81

^a Cut-off was determined by the point of intersection between the frequency distribution curves for the uninstructed and instructed criterion groups (Berk, 1976).

^b Percentage of uninstructed group who scored below the cut-off score.

^c Percentage of instructed group who scored above the cut-off score.

^d Percentage of both groups who scored above or below the cut-off score as their criterion group indicated they should.

Table 6. Cut-off scores set by steering committee for total CLAI and subtests and percentages of criterion groups classified as masters and nonmasters of the content

Section of test	Cut-off ^a score	Percentage of criterion group appropriately classified as masters and nonmasters of the content		
		Uninstructed ^b (N=110)	Instructed ^c (N=152)	Both groups ^d (N=262)
III. Systems	19	90	51	67
IV. Applications	18	90	51	67
V. Programming	11	85	54	67
Total test	47	92	54	70

^aCut-off was determined by the judgments of the steering committee to be the mean score for the instructed group.

^bPercentage of uninstructed group who scored below the cut-off score.

^cPercentage of instructed group who scored above the cut-off score.

^dPercentage of both groups who scored above or below the cut-off score as their criterion group indicated they should.

Table 7. Comparison of CLAI test scores of instructed versus uninstructed criterion groups

	Number	Mean	S.D.	T-Value	2-Tailed Probability ($p < .001$)
Section II					
Instructed	152	38.7	9.5	3.38	0.001**
Uninstructed	110	42.5	8.2		
Section III					
Instructed	152	18.6	3.8	-11.92	0.001**
Uninstructed	110	11.7	5.1		
Section IV					
Instructed	152	17.4	4.4	-11.23	0.001**
Uninstructed	110	10.8	5.1		
Section V					
Instructed	152	11.2	3.8	-10.16	0.001**
Uninstructed	110	5.7	4.6		
Total test					
Instructed	152	47.2	10.2	-12.58	0.001**
Uninstructed	110	28.2	13.2		

** $p < .01$.

Table 8. CLAI test-item analysis data from instructed and uninstructed criterion groups

Item	Comp ^a	I-C ^b	Uninstructed			Instructed			DIS ^f _{uigd}
			Omits ^c	Diff ^d	Disc ^e	Omits	Diff	Disc	
1	S1	4.6	19	53	.41	2	69	.32	16
2	S2	4.2	6	80	.21	2	91	.21	11
3	S3	4.0	7	74	.49	0	87	.33	13
4	S4	4.2	5	55	.36	0	70	.02	15
5	S4	4.6	6	64	.46	0	78	.22	14
6	S5	4.2	34	25	.27	2	41	.26	16
7	S6	4.6	22	36	.53	0	36	.33	0
8	S7	4.4	13	55	.62	0	82	.34	27
9	S8	-	8	63	.37	0	78	.15	15
10	S9	3.8	20	62	.60	0	89	.10	27
11	S9	4.2	15	53	.48	1	75	.32	22
12	S10	3.7	12	64	.37	0	56	.28	-8
13	S11	4.7	27	20	.20	1	20	.09	0
14	S12	4.6	24	42	.12	0	50	.16	8
15	S13	-	11	39	.23	0	62	.27	23
16	S14	-	11	30	.48	0	39	.18	9
17	S15	4.2	15	61	.63	1	77	.26	16
18	S16	4.4	4	55	.38	0	70	.42	15
19	S16	4.6	11	46	.39	2	77	.18	31
20	S17	4.4	38	24	.49	1	55	.25	31
21	S17	4.0	34	37	.54	0	61	.29	24
22	S18	3.0	17	55	.34	1	51	.09	4

^aCompetency that item measures. Code refers to the section of the test (System-S, Applications-A, Programming-P) and the number of the competency from the list of competencies in Appendix E.

^bItem-competency congruence (mean rating of judges on a scale of 1-5, 5=high congruence).

^cNumber of students who omitted the item.

^dDifficulty Index (percent of students who answered item correctly, 0-100).

^eDiscrimination Index (item-score correlation, -1.00-1.00).

^fUninstructed-Instructed Group Difference Discrimination Index ($Diff_I - Diff_U$, -100 - +100).

Table 8. (continued)

Item	Comp ^a	I-C ^b	Uninstructed			Instructed			DIS ^f _{uigd}
			Omits ^c	Diff ^d	Disc ^e	Omits	Diff	Disc	
23	S19	4.5	12	37	.58	0	43	.25	6
24	S20	-	31	16	.09	1	55	.51	39
	S21	-							
25	S20	-	21	47	.49	1	72	.42	25
	S21	-							
26	S22	4.8	15	46	.38	0	93	.31	47
27	S23	3.4	20	39	.47	0	63	.30	24
28	S24	-	9	19	.25	0	37	.17	18
29	S25	4.0	7	48	.30	0	92	.19	44
30	A1	3.0	11	47	.04	3	51	.20	3
31	A2	-	11	60	.48	2	55	.44	-5
32	A3	-	9	41	.29	1	64	.30	23
33	A4	4.0	4	87	.47	1	98	.28	11
34	A5	3.0	5	50	.40	3	76	.44	26
	A6	4.5							
35	A7	4.3	5	62	.57	1	87	.38	25
36	A7	4.0	5	21	.23	1	32	.23	11
37	A8	3.8	5	59	.41	1	75	.46	16
38	A9	4.8	9	71	.58	2	93	.25	22
39	A10	4.2	22	52	.60	2	81	.42	29
	A11	-							
40	A10	4.6	28	52	.77	2	81	.48	29
	A11	4.2							
41	A12	4.2	21	12	.32	1	82	.21	70
	A13	-							
42	A12	4.8	34	36	.36	2	89	.38	53
	A13	-							
43	A14	-	13	34	.59	1	30	.43	-4
	A15	-							
44	A14	-	12	54	.40	1	33	.30	-21
45	A16	4.3	12	38	.41	1	40	.36	2
	A18	-							
46	A16	-	15	75	.62	1	92	.36	17
	A17	3.6							
	A19	4.4							
47	A18	5.0	11	60	.43	1	62	.42	2
48	A20	4.0	15	17	.09	1	68	.37	49
49	A21	3.6	28	30	.56	1	58	.22	28
50	A22	-	20	59	.66	1	84	.37	25
51	A23	-	13	46	.28	1	48	.27	2

Table 8. (continued)

Item	Comp ^a	I-C ^b	Uninstructed			Instructed			DIS ^f _{uigd}
			Omits ^c	Diff ^d	Disc ^e	Omits	Diff	Disc	
52	A22	-	18	17	.07	2	29	.11	12
	A23	-							
	A25	-							
53	A24	5.0	12	41	.26	1	66	.23	25
	A25	3.8							
54	A24	4.6	16	47	.40	1	70	.42	23
	A25	4.0							
55	A24	-	21	31	.39	2	47	.40	16
	A25	-							
56	A24	-	14	13	.27	1	23	.15	10
	A25	-							
57	A25	-	24	22	.62	1	48	.40	26
58	P1	3.4	17	41	.37	1	44	.34	3
	P2	3.4							
59	P3	4.7	26	45	.75	2	67	.51	22
60	P3	4.6	30	25	.51	2	45	.26	20
61	P4	5.0	25	61	.48	3	64	.21	3
62	P5	5.0	25	71	.58	1	82	.37	11
63	P5	5.0	26	61	.73	1	73	.36	12
64	P6	4.6	34	36	.73	1	50	.50	14
65	P4	4.4	39	46	.66	7	42	.25	-4
	P7	-							
66	P8	3.5	37	15	.46	1	48	.32	33
67	P8	3.5	37	45	.63	1	63	.21	18
68	P9	5.0	39	27	.39	1	45	.22	18
69	P9	5.0	32	49	.67	1	56	.35	7
70	P10	3.8	28	16	.36	1	77	.52	61
71	P11	3.0	34	20	.41	1	25	.24	5
72	P2	4.4	39	21	.56	2	26	.20	5
	P7	-							
	P12	-							
	P14	-							
73	P15	4.6	41	38	.77	2	65	.25	27
74	P15	4.8	41	61	.89	2	80	.33	19
	P16	4.8							
75	P15	4.8	44	30	.58	2	30	.40	0
76	P15	4.0	42	41	.72	2	32	.39	-9
	P13	3.8							
77	P17	4.8	42	34	.46	2	31	.15	-3
78	P18	-	72	21	-.07	14	24	.04	3
79	P19	-	72	29	.68	15	37	.33	8
80	P20	-	74	22	.14	21	24	.19	4

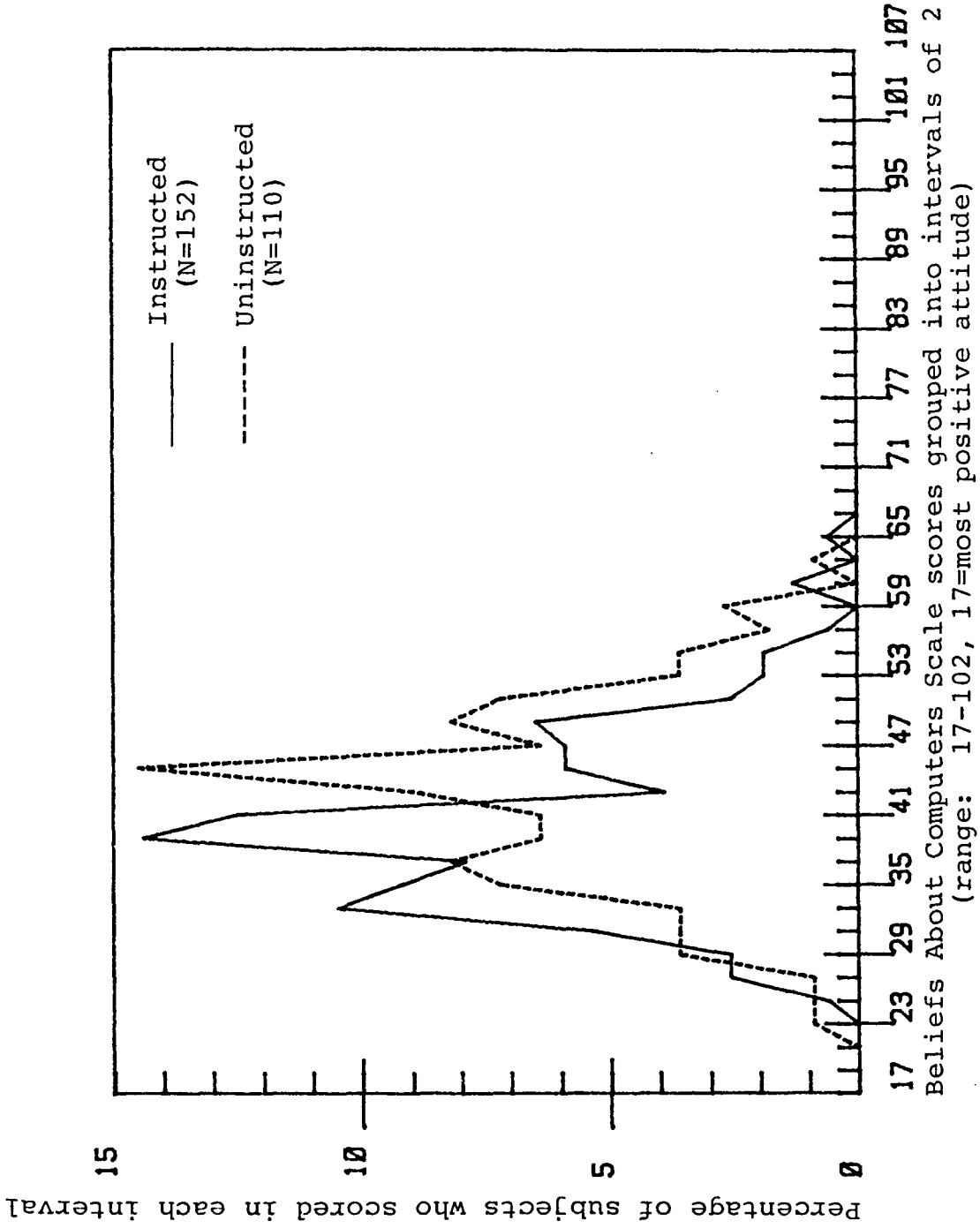


Figure 1. Frequency distributions of scores for CLAI Section II, Part One (Beliefs About Computers Scale) by criterion group

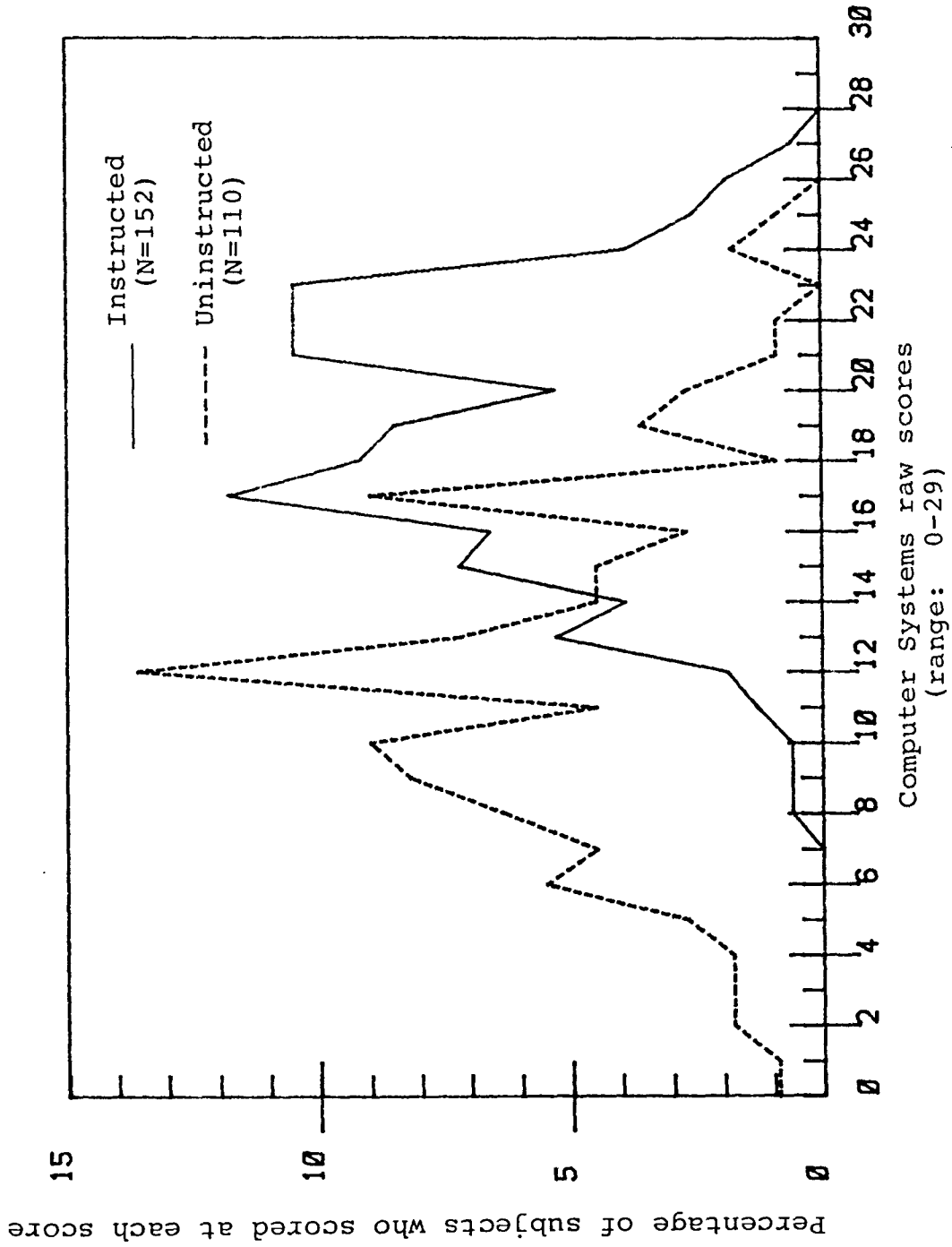


Figure 2. Frequency distributions of scores for CLAI Section III (Computer Systems) by criterion group

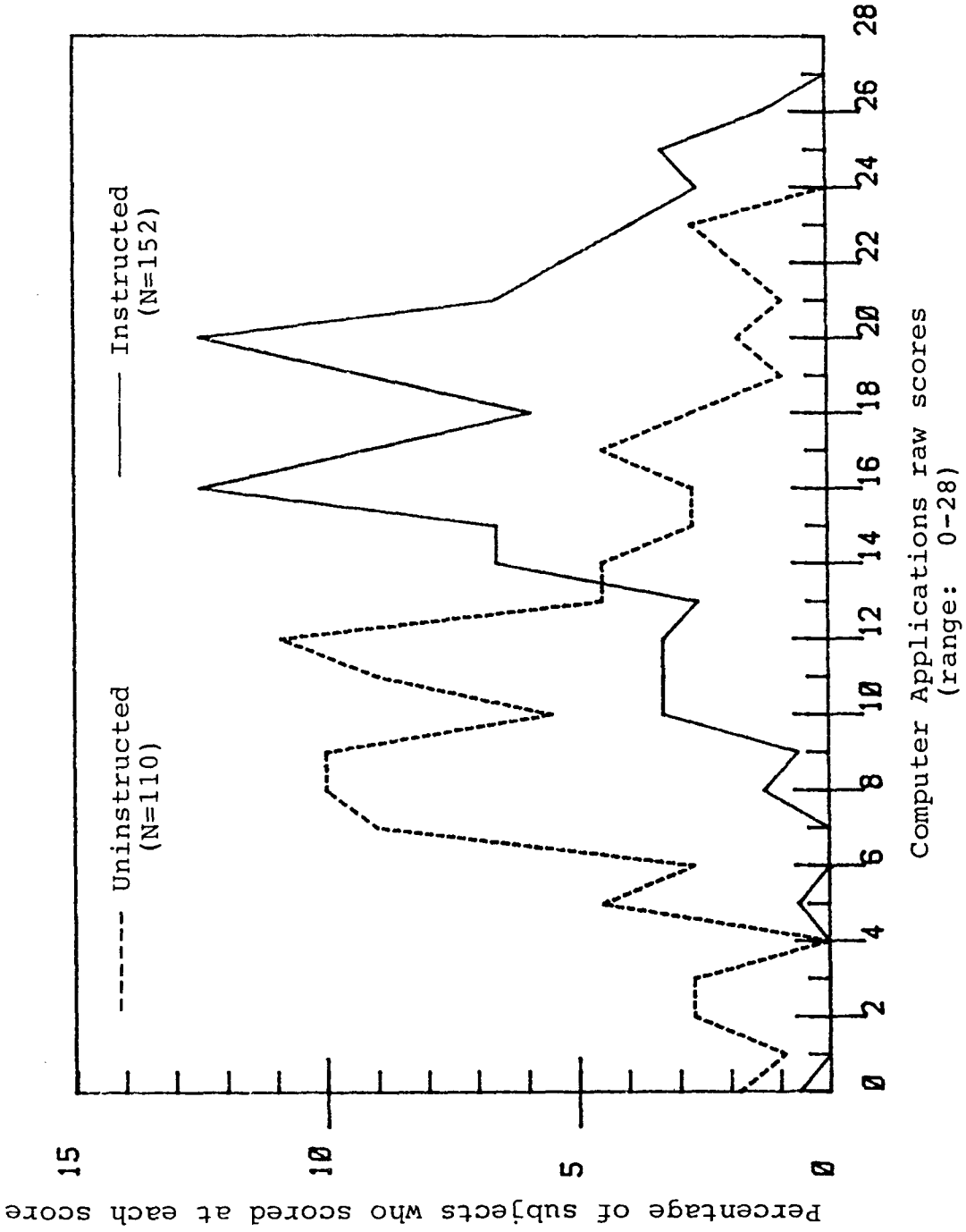


Figure 3. Frequency distributions of scores for CLAI Section IV (Computer Applications) by criterion group

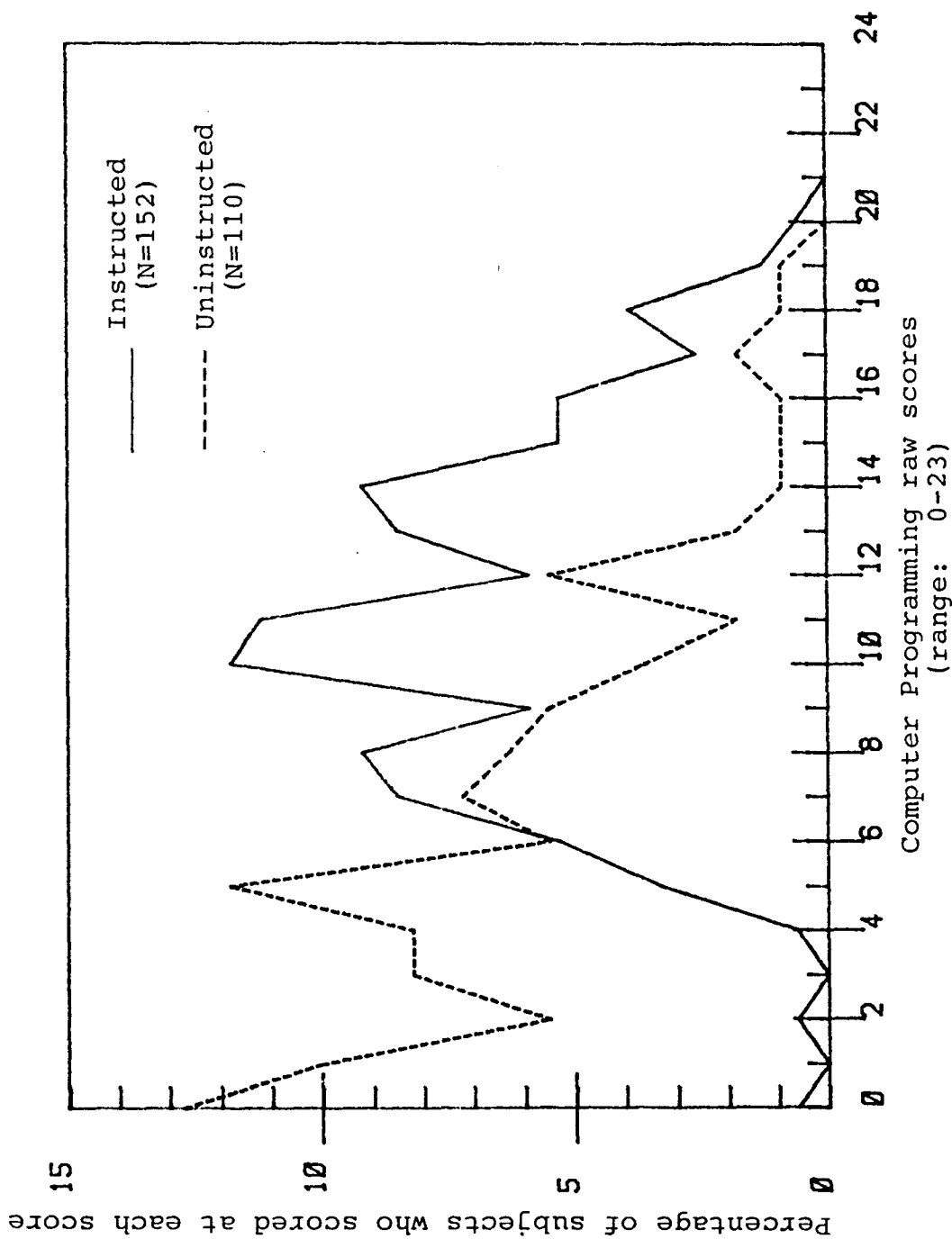


Figure 4. Frequency distributions of scores for CLAI Section V (Computer Programming) by criterion group

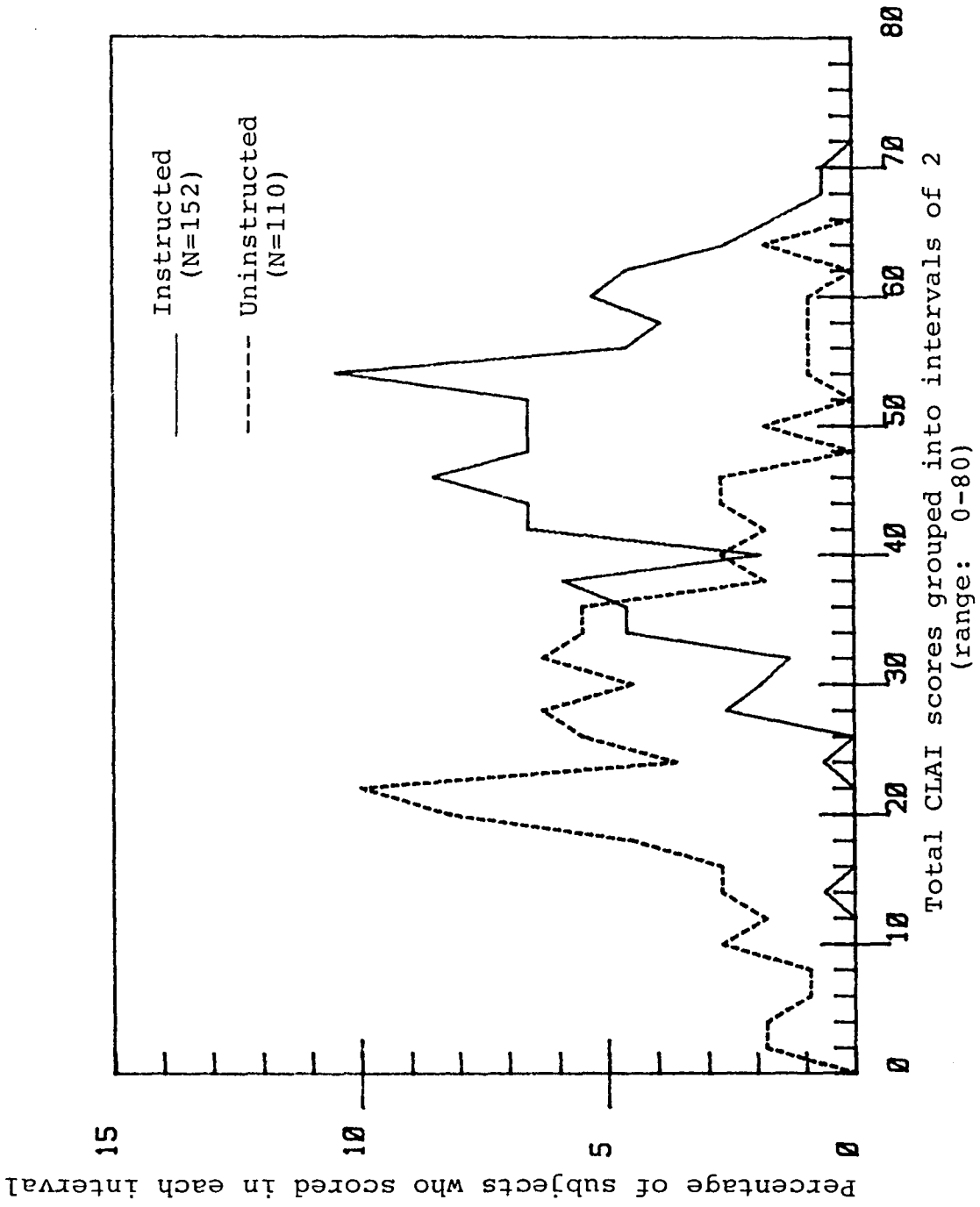


Figure 5. Frequency distributions of scores for total CLAI by criterion group

CHAPTER FIVE: DISCUSSION OF RESULTS

Educators have recently been faced with the task of meeting the demands of society to educate students so they become computer literate. Many educational institutions have instigated a computer literacy requirement for graduation. Public school districts are developing computer literacy programs and expecting students to be computer literate. However, there are currently no known standards for determining what aspects of computer literacy should be taught at various levels of education, and no effective means of measuring the level of computer literacy possessed at various levels of education. A measure of computer literacy is needed to facilitate the development of computer literacy courses that are appropriate to the needs of the students. Such an instrument would be critical for providing students with instruction and experiences suited to their level of computer literacy.

The purpose of this study was to construct a valid and reliable criterion-referenced instrument to assess the computer literacy (knowledge, skills, attitudes) of students. Three primary purposes of the Computer Literacy Assessment Instrument were identified.

1. To document the computing competencies of students and to collect normative data about

various subgroups of students.

2. To justify revisions in the content of introductory computing courses to reflect student needs and prerequisite skills.
3. To provide criteria for the placement of students in appropriate computing courses based on their level of computer literacy.

Development of the CLAI

An attempt was made to construct as valid and reliable an instrument as possible. The procedures followed in the development of the instrument were designed to contribute positively to the overall validity and reliability of the instrument and were very successful in producing such an instrument. The development of the test included the following steps.

First, a definition of computer literacy was developed based on an extensive review of the literature and consultations with computer specialists. Aspects of a wide variety of definitions supported in the literature were combined in an attempt to develop a representative definition of computer literacy (Chapter Three).

There has been much debate reported in the literature concerning the "right" definition of computer literacy, so not everyone will agree that the definition of computer literacy used in this study was the correct one. However,

the differences in the various definitions cited in the literature seem to result from differences in the purposes for using the word. The phrase "computer literacy" is being used in so many varying contexts that it is almost becoming a catch-all term for anything having to do with computers. It may be unfortunate that the domain of content measured by the instrument has been labeled "computer literacy" in view of the controversy surrounding the phrase. The use of another word may have resulted in more agreement concerning the actual definition. Nevertheless, "computer literacy" was the label assigned to the domain of content expressed in the operational definition in Chapter Three and measured by the completed instrument.

Second, a list of competencies were collected from computer specialists and judged to be representative of the domain of computer literacy as defined in Chapter Three. The competencies were identified for three cognitive subsections (computer systems, computer applications, and computer programming) and one affective subsection (computer attitudes) of the definition. The competencies were selected based on five major criteria: transferability within the domain, acceptability, terminality, transferability outside the domain, and ease of scorability.

Computer specialists then wrote test items designed to measure the competencies. Detailed domain specifications, including the competencies and guidelines for writing effective multiple choice questions, were used as a guide for the writing of the items.

Next, the items were pilot tested with two groups of students (instructed and uninstructed) to establish the content validity of the questions. An item analysis was conducted for the collection of item statistics and a judgmental item analysis was conducted to determine the item-competency congruence and the technical quality of the items. The item analysis and judgmental analysis information was used to identify the test items with the highest degrees of content validity and technical quality.

Finally, the revised instrument was administered to two criterion groups, college students who were either instructed or uninstructed in computer literacy as defined in Chapter Three. The results of the test were used to determine performance standards, or cut-off scores, for classifying subjects as computer literate or not computer literate and to collect reliability, validity, and normative data for the entire instrument and each of its subtests.

Reliability

The instrument as a whole was shown to be very reliable. The internal consistency reliability estimates for the two criterion groups were 0.86 ($r=0.86$) for the instructed group and 0.91 ($r=0.91$) for the uninstructed group. The internal consistency reliability estimates for the individual subtests were somewhat lower (Table 4), but still indicated that the subtests were reliable for at least one of the criterion groups. For an instrument with characteristics of both criterion-referenced and norm-referenced measurements, such as the CLAI, a reliability estimate of 0.70 or greater was desired. Two sections of the CLAI, Computer Systems and Computer Programming, had reliability estimates lower than 0.70 for the instructed group ($r=0.64$ and $r=0.69$ respectively) and higher than 0.70 for the uninstructed group ($r=0.78$ and $r=0.81$ respectively).

The Kuder-Richardson 20 formula that was used to calculate the reliability estimate requires the following data: the number of items in the test, the standard deviation of the test scores, and the difficulty of each item in the test or the average difficulty of all items as reflected in the mean test score. The lower reliability estimates for Sections III and V for the instructed group

may be attributed to items with inappropriate difficulty indices. An examination of the item analysis may reveal items that could be eliminated or revised to improve the reliability of these subtests for the instructed group. The lower reliability estimates may also be attributed to the fact that these subtests contained fewer items and had lower variances than the total test.

The reliability estimates for Sections III, IV, V and the total instrument for the uninstructed group were all higher than the reliability estimates for the instructed group. This may be attributable to the fact that many of the uninstructed students consistently omitted, or answered incorrectly, a large number of the items. This increased the internal consistency of the test scores for the uninstructed group.

Section II, Part One (BACS) (Ellsworth and Bowman, 1982) was shown to have internal consistency reliability estimates of 0.81 ($r=0.81$) for the instructed group and 0.73 ($r=0.73$) for the uninstructed group. This finding supported Ellsworth and Bowman's findings. Ellsworth and Bowman (1982) found the BACS to have an internal consistency reliability of 0.77 ($r=0.77$) for a sample of students who varied in their exposure to computers.

Validity

Construct validity was established by demonstrating that the mean score of the instructed students was higher than the mean score of the uninstructed students and that the difference did not occur by chance. The mean scores of the instructed group for Sections III, IV, V, and the total test were all between one and two standard deviations above the mean scores of the uninstructed group for the same sections of the test. The differences between the mean scores for these sections were all found to be highly significant ($p < .001$). This indicated that there was less than one chance in one thousand that the differences occurred by chance.

The mean score of the instructed group for the BACS (Ellsworth and Bowman, 1982) was less than one standard deviation above the mean score of the uninstructed group. However, this difference was also found to be highly significant ($p < .001$).

Decision validity was established for the total test and its subtests by demonstrating that a high percentage of the decisions made to classify the students as nonmasters or masters of the content were appropriate decisions based on whether the students were instructed or uninstructed in computer literacy. Performance standards (cut-off scores)

were established for each section of the test using two different methods. The cut-off scores were the scores used to classify students as masters or nonmasters of the content. In other words, students scoring above the cut-off score were classified as masters and students scoring below the cut-off score were classified as nonmasters. It was expected that a large number of uninstructed students would be classified as nonmasters and a large number of instructed students would be classified as masters.

Using Berk's method (1976), the cut-off score for the total test was set at 37. A cut-off score of 37 resulted in a large percentage of the instructed students (84%) classified as masters of computer literacy and a lower percentage of the uninstructed students (78%) classified as nonmasters of computer literacy. The total percentage of students in both groups who were appropriately classified was 81%. However, a cut-off score of 37 was judged by the steering committee to be an unrealistically low criterion for classifying the students as computer literate.

The steering committee judged a more realistic cut-off score for the total test to be the mean score for the instructed group (47). A cut-off score of 47 resulted in 54% of the instructed group classified as computer literate and 92% of the uninstructed group classified as noncomputer

literate. The total percentage of students in both groups who were appropriately classified was 70%. These percentages appeared to be more representative of the actual characteristics of the two criterion groups than the percentages derived using Berk's (1976) method.

The discrepancy between the performance standards or cut-off scores set with the Berk (1976) method and those recommended by the steering committee may be explained by the characteristics of the students in the two criterion groups. While the students in the uninstructed group had not been specifically instructed in computer literacy as identified by the competencies, they did have varying levels of computer knowledge or experience, and were not totally uninstructed in computer literacy. The students in the instructed group also had varying levels of computer knowledge and experience. While the majority of the students in the instructed group were more instructed in computer literacy than the majority of the uninstructed group, they were not all knowledgeable to the same degree. As a result, the frequency distribution curves for the two groups overlapped more than they would have if the two groups had been truer to Berk's recommendations (1976) in terms of level of instruction.

Regardless of the method used to set the cut-off scores, the percentages of the students that were

appropriately classified as nonmasters or masters of computer literacy were high enough to indicate a high degree of decision validity. Future testing of the instrument using criterion groups that are more representative of totally uninstructed and instructed students would probably yield more appropriate cut-off scores and a more accurate estimate of decision validity. Further research may also be necessary to identify a more appropriate method for setting standards.

Normative Data

Normative data were collected for two groups, college students who were instructed in computer literacy concepts and college students who were uninstructed in computer literacy concepts. The means, standard deviations, frequency distributions, and percentile scores for all sections of the test for each norm group were reported in Chapter Four. The normative data can be used to make comparisons with subsequent CLAI scores. Normative data should be collected from other norm groups in future testings of the instrument. A number of interesting observations can be made concerning the norm data for the instructed and uninstructed groups.

First, both the instructed and uninstructed groups had positive attitudes toward computers. On a scale of 17-102, with 17 indicating the most positive attitude, the majority of the subjects in both groups scored on the positive side of the scale. While the instructed group had slightly more positive attitudes towards computers than the uninstructed group, the difference was not as great as was expected. It was thought that the uninstructed group would have much less positive attitudes towards computers because of misconceptions resulting from a lack of instruction. This was not the case, however. The increase in the use of computers in many facets of society during recent years may have contributed to the development of positive attitudes towards computers among computer users as well as nonusers.

Second, the frequency distributions for both criterion groups on all sections of the test resembled a normal distribution except the distribution for the uninstructed group on the Computer Programming section. This distribution was skewed toward the low end of the scale. Almost thirteen percent of the uninstructed group scored a zero on the Computer Programming section, while no more than two percent of the uninstructed group scored a zero on any other section. This may have indicated that the uninstructed group (in general) knew less about computer programming than computer systems or computer applications.

The type of computer experiences that the uninstructed group had been exposed to in situations other than direct instruction may have contributed to learning more information about computer systems and applications than computer programming. In other words, societal influences may have influenced the uninstructed subjects' knowledge about computer systems and computer applications more than their knowledge about computer programming.

The mean score for the instructed group for the total test was 47 out of 80 which is 59% of the items. This relatively low mean might indicate that the difficulty level of the instrument was too low for computer literate individuals. It might also indicate that the instructed group had a lower level of computer literacy than was hoped for. Considering the variety of instruction and experiences that the students in the instructed group had participated in, and the fact that many of them had only taken one computer literacy course, the latter explanation seems the most likely.

Possible Uses of the CLAI

There are several possible uses for the Computer Literacy Assessment Instrument (CLAI). The primary goal for the CLAI was to assess the computer literacy of

students at various educational levels, and to guide the development of appropriate instruction so all students can become computer literate. This primary goal could be accomplished by using the CLAI in the following ways.

Cut-off scores could be established in order to identify various degrees of computer literacy. Student scores could be compared to these cut-off scores, and could then be used as one of the criteria for assigning students to courses appropriate to their degree of computer literacy. A cut-off score could be identified as one of the criteria necessary for students to test out of a particular course or segment of a course. For example, a student who scored above the cut-off score on the computer systems and computer applications section of the CLAI, but below the cut-off score on the computer programming section could be placed in a course dealing strictly with programming.

The CLAI could be used to identify the specific computer literacy competencies possessed and not possessed by individuals in a particular course or group. The test items on the instrument were keyed to the competencies, so an examination of individual responses to each item could provide appropriate diagnostic information. Instruction could then be designed, or revised, to facilitate the students' mastery of the competencies that were lacking.

The CLAI could be used to identify persons who were highly computer anxious or who had highly negative attitudes towards computers. Anxiety or negative feelings towards computers may inhibit the subjects' interaction with computers and consequently influence their level of computer literacy. Persons with these characteristics may benefit from treatments designed to decrease their anxiety or improve their attitudes prior to receiving instruction about the cognitive components of computer literacy.

Suggestions for Future Research

A careful examination of the Computer Literacy Assessment Instrument and the results of its first administration to students revealed that several revisions could be made to improve the quality of the instrument. Further research could also be conducted that would contribute to the usefulness of the test and the quality of test score interpretations.

An examination of the test items themselves revealed several items in need of revision because of gender bias. The faulty items depict males and females in stereotypical roles that are not accurate reflections of the capabilities of both sexes. The items should be reworded to eliminate all references to gender. Words such as "an individual",

"a teacher", or "a scientist" could be used instead of gender-specific names, such as "Ms. Jones", or "Sam Smith".

The demographic information about the students obtained through Section I, Background Information, could be analyzed in relation to the test scores. The relationships between such variables as sex, age, educational level, educational major, grade point average, size of school, economic background, and previous computer experience, and CLAI scores could be determined using correlation coefficients. An analysis of variance could be conducted to determine if there are any significant interactions between the variables and the test scores. In order to examine the relationship between variables such as those mentioned above and the test scores, Section I, Background Information, might need to be revised to include items dealing with all of the desired variables. Some of the existing items might need to be reworded to yield more specific information.

The reliability coefficients for the Computer Systems ($r=0.64$) and Computer Programming ($r=0.69$) sections of the test for the instructed group should be improved. An examination of the item analysis data in Table 8 in Chapter Four may reveal some items with inappropriate difficulty and discrimination indices. Revision or deletion of the

faulty items might improve the internal consistency reliability estimate.

The test/retest reliability of the instrument needs to be determined to indicate the consistency of the test scores over time. Parallel versions of the instrument could be administered to the same group of students at different times to determine if the scores were consistent.

The instrument should be administered to criterion groups that are more comprehensively instructed (computer literate) and uninstructed (noncomputer literate) so that more appropriate and accurate cut-off scores could be established. This would also yield a more accurate estimate of construct and decision validity than was determined in this study.

An external content validation study needs to be conducted using a diverse sampling of computer specialists in order to determine whether the competencies and test items are representative of the domain of computer literacy and whether the items actually measure the competencies as accurately as they seem. Content validity was established in this study through the use of computer specialists, but only content specialists from Iowa State University participated. A broader sample of computer specialists would be valuable in order to attain a more comprehensive perspective of the content validity of the CLAI.

The CLAI should be administered to a variety of norm groups for the collection of more comprehensive norm data. Possible norm groups could be high school students, preservice teachers, inservice teachers, computer scientists, computer professionals, businesspersons, college students in various majors, and noncomputer professionals.

Summary of Results

1. The CLAI was found to be a valid measure of computer literacy.
2. The CLAI was found to be a reliable measure of computer literacy.
3. The CLAI could be used as one of the criteria to classify students as computer literate or noncomputer literate.
4. The CLAI could be used to facilitate decisions about the placement of students in computer literacy courses.
5. The CLAI could be used to document the computer literacy competencies of students, and could be used in identifying instructional needs and goals.
6. The CLAI could be used as a basis for the design and/or revision of appropriate computer literacy instructional programs.

REFERENCES

- Ahl, D. Survey of public attitudes toward computers in society. The Best of Creative Computing (Vol. 1). Morristown, New Jersey: Creative Computing Press, 1976.
- AIFPS and Time, Inc. A national survey of the public's attitudes towards computers. Arlington, Virginia: AFIPS Press, 1971.
- Anderson, R. E. National computer literacy, 1980. In Seidel, R. J., Anderson, R. E., & Hunter, B. (Ed.), Computer literacy: Issues and directions for 1985. New York: Academic Press, 1982.
- Anderson, R. E., & Klassen, D. L. A conceptual framework for developing computer literacy instruction. AEDS Journal, Spring 1981, 14(3), 128-150.
- Anderson, R. E., Krohn, K., & Sandman, R. User guide for the Minnesota Computer Literacy and Awareness Assessment. St. Paul, Minnesota: Minnesota Educational Computing Consortium, 1980.
- Ball, F. H., & Chapp, S. Be a computer literate. Morristown, New Jersey: Creative Computing Press, 1977.
- Berk, R. A. Determination of optimal cutting scores in criterion-referenced measurement. Journal of Experimental Education, 1976, 45, 4-9.
- Berk, R. A. A critical review of domain specification/item generation strategies for criterion-referenced tests. Paper presented at the meeting of the American Educational Research Association, April 1979. ERIC ED 170 382.
- Berk, R. A. (Ed.). Criterion-referenced measurement: The state of the art. Baltimore: Johns Hopkins University Press, 1980.
- Berk, R. A. Item analysis. In Berk, R. A. (Ed.), Criterion-referenced measurement: The state of the art. Baltimore: Johns Hopkins University Press, 1980.
- Billings, K., & Moursund, D. Are you computer literate? Forest Grove, Oregon: Dilithium Press, 1979.

- Bitter, G. G. The road to computer literacy: A scope and sequence model. Electronic Learning, September 1982, 2(1), 60-63.
- Bloom, F. S., Englehart, M. D., Furst, E. J., Hill, W. H., & Krathwohl, D. R. Taxonomy of educational objectives: Cognitive domain. New York: David McKay, 1956.
- Carpenter, T. P., Corbitt, M. K., Kepner, H. S., Lindquist, M. M., & Reys, R. E. The current status of computer literacy: NAEP results for secondary students. Mathematics Teacher, December 1980, 73(9), 669-673.
- Conference Board of the Mathematical Sciences. Recommendations regarding computers in high school education. Washington, D. C.: Author, 1972.
- Cronbach, L. J. Essentials of psychological testing (3rd ed.). New York: Harper and Row, 1970.
- Dennis, J. R. Undergraduate programs to increase instructional computing in schools. Proceedings of the 1978 Conference on Computers in the Undergraduate Curricula: CCUC9, Denver, June 1978.
- Ebel, R. L. Essentials of educational measurement. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1972.
- Educational Testing Service. Multiple choice questions: A close look. Princeton, New Jersey: Educational Testing Service, 1973. ERIC ED 081 783.
- Eisele, J. E. A case for universal computer literacy. Journal of Research and Development in Education, 1980, 14(1), 84-89.
- Ellsworth, R., & Bowman, B. E. A "beliefs about computers" scale based on Ahl's questionnaire items. The Computing Teacher, December 1982, 10(4), 32-34.
- Fremer, J. Handbook for conducting task analyses and developing criterion-referenced tests of language skills. (PR 74-12). Princeton, New Jersey: Educational Testing Service, 1974.
- Friedman, D. Specific areas of national need in educational computing. Educational Computing Magazine, January-February 1983, 3(1), 26-27.

- Glaser, R. Instructional technology and the measurement of learning outcomes. American Psychologist, 1963, 18, 519-521.
- Glaser, R., & Nitko, A. J. Measurement in learning and instruction. In Thorndike, R. L. (Ed.), Educational measurement (2nd ed.). Washington: American Council on Education, 1971.
- Gray, W. M. A comparison of Piagetian theory and criterion-referenced measurement. Review of Educational Research, Spring, 1978, 48(2), 223-49.
- Haladyna, T., & Roid, G. The role of instructional sensitivity in the empirical review of criterion-referenced test items. Paper presented at the annual meeting of the American Educational Research Association, April 1980. ERIC ED 185 100.
- Hambleton, R. K. Test score validity and standard-setting methods. In Berk, R. A. (Ed.), Criterion-referenced measurement: The state of the art. Baltimore: Johns Hopkins University Press, 1980.
- Hambleton, R. K., & Eignor, D. R. Guidelines for evaluating criterion-referenced tests and test manuals. Journal of Educational Measurement, Winter 1978, 15(4), 321-327.
- Hambleton, R. K., & Eignor, D. R. A practitioner's guide to criterion-referenced test development, validation, and test score usage. Laboratory of Psychometric and Evaluative Research Report no. 70. Amherst, Massachusetts: School of Education, University of Massachusetts, 1979.
- Hambleton, R. K., & Novick, M. R. Toward an integration of theory and method for criterion-referenced tests. Journal of Educational Measurement, Fall 1973, 10(3), 159-170.
- Hambleton, R. K., Swaminathan, H., Algina, J., & Coulson, D. B. Criterion-referenced testing and measurement: A review of technical issues and developments. Review of Educational Research, Winter 1978, 48(1), 1-47.

- Hansen, T. P., Klassen, D. L., Anderson, R. E., & Johnson, D. C. What teachers think every high school graduate should know about computers. School Science and Mathematics, October 1981, 81(6), 467-472.
- Harris, C. W., Alkin, M. C., & Popham, W. J. Problems in criterion-referenced measurement. CSE monograph series in evaluation, No. 3. Los Angeles: Center for the Study of Evaluation, University of California, 1974.
- Hemphill, J., & Westie, C. M. The measurement of group dimensions. Journal of Psychology, 1950, 29, 325-342.
- Hively, W., Patterson, H. L., & Page, S. H. A "universe-defined" system of arithmetic achievement tests. Journal of Educational Measurement, Winter 1968, 5(4), 275-290.
- Hunter, B. An approach to integrating computer literacy into the K-8 curriculum. Paper presented at the National Computer Conference, June 1980. ERIC Ed 195 247.
- Hunter, B. What makes a computer-literate college? Proceedings of the 1978 Conference on Computers in the Undergraduate Curricula: CCUC9, Denver, June 1978.
- Hunter, B. Definition and conceptual structure for computer literacy. Research report: Human Resources Research Organization under a subcontract with Educational Testing Service (NIE contract no. 400-82-0024), January, 1983.
- Iowa State University Test and Evaluation Services. Summary of services: Test and evaluation services. Unpublished manuscript. Iowa State University, 1983.
- Johnson, D. C., Anderson, R. E., Hansen, T. P., & Klassen, D. L. Computer literacy - What is it? Mathematics Teacher, February 1980, 73(2), 91-96.
- Kelman, P. (Ed.). The 1983 classroom computer news directory of educational computing resources. Watertown, Massachusetts: Intentional Educations, Inc., 1983.

- Klassen, D. L., Anderson, R. E., Hansen, T. P., & Johnson, D. C. A study of computer literacy in science education: Final report. 1978-1980. St. Paul: Minnesota Educational Computing Consortium, 1980.
- Klein, S. P., & Kosecoff, J. Issues and procedures in the development of criterion referenced tests. 1973. ERIC document ED 083 284.
- Kull, J. A. & Archambault, F. X. A survey of teacher preparation in computer education. Journal of Teacher Education, January-February 1984, 25(1), 16-19.
- Lichtman, D. Survey of educators' attitudes toward computers. Creative Computing, 1979, 5, 48-50.
- Logsdon, D. M. An experimental comparison of the effectiveness of two domain definition strategies for criterion-referenced tests. Paper presented at the annual meeting of the Eastern Educational Research Association, Philadelphia, March 1981. ERIC ED 206 714.
- Luehrmann, A. Computer literacy - What should it be? Mathematics Teacher, December 1981, 74(9), 682-686.
- Masat, F. E. Computer literacy in higher education. AAHE-ERIC/Higher Education Research Report No. 6, 1981. Washington, D.C.: American Association for Higher Education, 1981.
- Mason, E. J., & Bramble, W. J. Understanding and conducting research/Applications in Education and Behavioral Sciences. New York: McGraw-Hill Book Company, 1978.
- Maurer, M. M. Development and validation of a measure of computer anxiety. Unpublished Master's Thesis. Iowa State University, 1983.
- Millman, J. Criterion-referenced measurement. In Popham, W. J. (Ed.), Evaluation in education: Current applications. Berkeley, California: McCutchan Publishing Company, 1974.
- Molnar, A. R. The next great crisis in American education: Computer literacy. Paper presented at the annual meeting of the Society for Applied Learning Technology. February, 1978. ERIC ED 191 733.

- Molnar, A. R. The coming of computer literacy: Are we prepared for it? Educational Technology, January 1981, 21(1), 26-28.
- Morris, L. L., & Fitz-Gibbon, C. T. How to measure achievement. Beverly Hills: Sage Publications, 1978.
- Moursund, D. What is computer literacy? Creative Computing, November-December 1976, 2(6), 55.
- Moursund, D. International Council for Computers in Education organization members. The Computing Teacher, April 1983, 10(8), 80.
- Moursund, D. Summer school course listings: Programming a great summer. The Computing Teacher, March 1982, 9(7), 12-23.
- National Commission on Excellence in Education. An open letter to the American people. A nation at risk: The imperative for educational reform, Education Week, April 27, 1983, II(3), 12-16.
- Poggio, J. P., Glassnap, D. R., & Eros, D. S. An empirical investigation of the Angoff, Ebel and Nedelsky standard setting methods. Paper presented at the annual meeting of the American Educational Research Association, Los Angeles, April 1981. ERIC ED 205 552.
- Popham, W. J. (Ed.). Educational evaluation. Englewood Cliffs, New Jersey: Prentice-Hall, 1975.
- Popham, W. J. Criterion-referenced measurement. Englewood Cliffs, New Jersey: Prentice-Hall, 1978.
- Popham, W. J. Domain specification strategies. In Berk, R. A. (Ed.), Criterion-referenced measurement: The state of the art. Baltimore: The Johns Hopkins University Press, 1980.
- Popham, W. J., & Husek, T. R. Implications of criterion-referenced measurement. Journal of Educational Measurement, 1969, 6(1), 1-9.
- Rahmlow, H. F., & Woodley, K. K. Objectives-based testing: A guide to effective test development. Englewood Cliffs, New Jersey: Educational Technology Publications, 1979.

- Rohner, D. J. Development and validation of an index of computer anxiety among prospective teachers. Unpublished Master's Thesis. Iowa State University, 1981.
- Roid, G., & Haladyna, T. M. A comparison of several linguistic-based, multiple choice item writing algorithms. Paper presented at the annual meeting of the American Educational Research Association, Toronto, March 1978.
- Rovinelli, R. J., & Hambleton, R. K. On the use of content specialists in the assessment of criterion-referenced test item validity. Paper presented at the annual meeting of the American Educational Research Association, San Francisco, 1976.
- Ryan, J. J. Teacher judgments of test item properties. Journal of Educational Measurement, 1968, 5, 301-306.
- Seidel, R. J., Anderson, R. E., & Hunter, B. (Ed.). Computer Literacy: Issues and directions for 1985. New York: Academic Press, 1982.
- Smith, J., & Moum, G. S. Proceedings of NECC-82 National Educational Computing Conference 1982. Columbia, Missouri: The University of Missouri, 1982.
- Ward, C. Preparing and using objective questions. Cheltenham, England: Stanley Thornes (Publishers) Ltd., 1981.
- Watt, D. H. Computer literacy: What should schools be doing about it? Classroom Computer News, November-December 1980, 1(2), 1, 26.

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to many individuals for their assistance with this thesis and my entire graduate program. Each of these persons offered his or her own unique form of assistance, all of which was accepted and valued. I wish to thank each of them personally.

First, I would like to thank my major professor, Dr. Michael Simonson, for his professional guidance, encouragement, and patience. I especially thank him for his confidence in my ability to succeed. He had a special talent of instilling that confidence in me, so that I too felt that I would succeed.

I also wish to thank the other members of the two committees that guided my work: Drs. Elaine Jarchow and Dahlia Stockdale for their support on my Program of Study committee, and Drs. Michael Simonson, Roger Volker, and Ann Thompson for their helpful contributions on the CLAI steering committee. I am also grateful to the Research Institute for Studies in Education (RISE) for their support, both financially and professionally.

A number of fellow graduate students and faculty were also a tremendous help to me. They were always there when I needed them: to volunteer to write test items, to serve

as computer specialists during the judgmental item analysis, to offer encouragement, a good laugh, or a shoulder to cry on, but most importantly, to be my friends. Thank you Trish, Annette, Cindi, Meg, De, Kathy, Cym, Fred, Greg, Mike, Chris, Jane, Dr. Simonson, Dr. Thompson, and Dr. Volker for making the last two years such a memorable experience. I will miss you all.

Most of all, I wish to extend a special thank you to three very special persons who were always by my side (in spirit, if not always in person). I sincerely thank my parents, Richard and Patty Montag for their continual love, guidance, encouragement, patience, and support. I especially thank my best friend and fiancé, Charlie Torardi, for his love, his friendship, his encouragement, and for being my inspiration.

APPENDIX A:
SURVEY OF COMPUTER LITERACY COMPETENCIES

August 24, 1983

Mary Montag
Instructional Resources Center
N031 Quadrangle
Iowa State University
Ames, Iowa 50011

Mr. Glenn Fisher
School Computer Specialist
Alameda County School District
313 W. Winton
Hayward, California 94544

Dear Mr. Fisher,

As a graduate student at Iowa State University in the department of Curriculum and Instructional Media in the college of Professional Studies in Education, I am involved in a project to construct and validate an instrument to measure the level of computer literacy possessed by incoming undergraduate students in education. The project is being funded by the Research Institute for Studies in Education (RISE) and is directed by a steering committee of Dr. Michael Simonson, Dr. Roger Volker, and Dr. Ann Thompson, professors in Professional Studies in Education and Instructional Computing.

The instrument will have three primary purposes: 1) to document the computing competencies of incoming students, 2) to allow for revisions in the content of introductory computing courses to reflect student needs, and 3) to provide for the appropriate placement of students in computing courses.

The steering committee and I have developed a definition of computer literacy that has four major subsections (see next page). This four part definition will be used as a guide in the development of computer literacy competencies. The construction of specific test items will be based on the computer literacy competencies that will be identified.

Your assistance, Mr. Fisher, in the identification of appropriate computer literacy competencies based on our definition will greatly contribute to the quality and validity of the computer literacy competencies and the subsequent test items. We are asking you to please read our definition and explanation of the four subsections (next page) and write two (or more) computing competencies for each subsection that you believe should be included in our list of computer literacy

competencies. Please write any competency that you believe to be important even if it does not pertain to one of the subsections we have identified. We are very interested in obtaining the most comprehensive list of computer literacy competencies as possible.

Please write the competencies on the form provided and return it to me in the enclosed envelope as soon as possible. Any additional information you may wish to contribute to this project will be greatly appreciated (e.g. computer course outlines, objectives, and tests). Mr. Fisher, thank you very much for your valuable assistance and cooperation in this project. I am looking forward to receiving your response in the near future.

Sincerely yours,

Mary Montag
Graduate assistant

DEFINITION OF COMPUTER LITERACY

Computer literacy is defined as an understanding of computer characteristics, capabilities, and applications, and the ability to implement this knowledge in the skillful, productive use of computer applications suitable to individual roles in society. The knowledge and skills of computer literacy as defined above can be divided into four major categories. The test of computer literacy will have these four sections.

1. COMPUTER SYSTEMS:

Knowledge and awareness of computer systems (mainframe, mini and micro computers) including historical development, terminology, identification of computer hardware and software, the relationship between hardware and software, the operation of computer systems, and the relationship between different computer systems.

2. COMPUTER APPLICATIONS:

Knowledge and skills in the applications of computers including: the general present applications of computers in society, future implications of computers in society, the limitations of computer applications, and the ability to evaluate, select, and implement a variety of specific computer applications and software packages (e.g., data base management, word processing, financial packages, authoring languages, programming, graphics, sound synthesizers, problem solving, computer-managed instruction, and computer-assisted instruction).

3. COMPUTER PROGRAMMING:

Knowledge and skills in programming computers including problem solving strategies, algorithm and flowcharting skills, structure, logic, high-level computer languages, and languages of specific software packages.

4. COMPUTER ATTITUDES:

Positive attitudes towards computers, their capabilities, and their impact on society, as well as an anxiety-free desire to use computers.

August 24, 1983

To ensure the accuracy of our documentation, please provide the following information and return this form with your list of competencies.

Date: _____

Name: _____

Title or
position: _____

Institution: _____

Address: _____

City, state, zip: _____

Please return to:

Mary Montag
Instructional Resources Center
N031 Quadrangle
Iowa State University
Ames, Iowa 50011

COMPUTER LITERACY COMPETENCIES

Please read the definition of computer literacy on the preceding page, and write two or more computer literacy competencies for each of the four subsections below, and others if necessary.

1. COMPUTER SYSTEMS:

2. COMPUTER APPLICATIONS:

3. COMPUTER PROGRAMMING:

4. COMPUTER ATTITUDES:

5. OTHER:

APPENDIX B:
THE COMPLETE UNREVISED LIST
OF COMPUTER LITERACY COMPETENCIES

THE UNREVISED LIST OF COMPUTER LITERACY COMPETENCIESKey:

- * Competency identified from the literature.
- (#) Competency identified from the Computer Literacy Competencies Survey. The number in parentheses is the number of survey respondents who mentioned the competency.

The numbers in parentheses following each competency are the identification numbers of the specific respondents who mentioned each competency.

COMPUTER SYSTEMS:

- (1) Computer systems refer to the equipment and programs necessary for computer applications: Historical development, computer hardware, systems configuration, computer software, computer glossary, and the operation of computers. (74)
- * (2) Explain the binary system and its functional relationship to digital computers. (10,78)
- * (3) Explain the process of data processing in a computer (the transformation of data by means of a set of pre-defined rules and computers process data by searching, sorting, deleting, updating, summarizing, and moving). (61,82,51)
- * (3) Explain the primary function of a computer system as the input of information, processing of information, and output of information. (71,50,51)
- * (5) Explain that a computer is an impersonal, literal machine (hardware) incapable of functioning without a set or program of instructions written in programming language, and that such programs or sets of instructions are called software. (A computer is capable of storing data and programs). (3,21,13,69,81)
- * (10) Understand the function of an operating system and be able to compare and contrast the characteristics of different operating systems. (2,40,63,70,5,49,50,81,9,13)
- * Distinguish between and list advantages of centralized data processing and distributed data processing.
- * Compare computer processing and storage capabilities to the human brain listing some general similarities and differences.

- (2) Explain how high level languages are transformed to machine understandable code (data representation: Ebedic, Ascii code). (38,44)
- (1) Describe the computer as a problem solving machine. (17)
- * Identify and distinguish between analog and digital computing operations.
 - (1) Describe how a "biochip" works. (45)
- * (3) Understand the capabilities and limitations of computer hardware. (51,82,79)
- * (4) Explain the characteristics (size, limitations, differences and functions) of RAM, ROM, EPROM, floppy disc memory, hard disc memory, and bubble memory. (43,32,19,81)
- * (22) Understand and use computer terminology: hardware, software, CPU, memory, input, output, network, compiler, interpreter, machine language, bit, byte, kilobyte, K, on-line, off-line, NIBBLE, hexa-decimal, chip, mainframe, microprocessor, DOS, binary, time-share, RAM, ROM, BASIC, vacuum tube, transistor, integrated circuit, I/O, bug, peripheral, disk, disk drive, printer, keyboard, microprocessor, program, hardcopy, CRT, CAI, CMI, modulator etc. (6,7,10,12,17,21,27,39,41,51,52,54,57,73,25,32,78,80,81,82,87,88)
- * (37) Identify and understand (explain) the function of the major components of a computer system: input, output, CPU (control unit, arithmetic/logic unit), storage, and memory. (1,2,3,7,11,12,16,17,21,22,23,24,29,31,33,38,39,44,47,50,51,52,54,57,58,60,66,69,70,71,72,78,81,82,84,67,85,86,88)
- * (6) Identify and explain the purpose/function of the control unit and the arithmetic/logic unit of the CPU of a microprocessor. (43,2,29,81,82,35)
 - (1) From a list of major computer system components, name the components that are essential for specific instructionally related tasks. (67)
 - (1) Identify/describe the input and output devices for the computer fuel regulation system for the current American automobiles. (45)
- * Identify and distinguish between special purpose and general purpose computers.
 - (1) Go easy on terminology requirement-it's a serious part of the present problem-use terminology only where it communicates clearly rather than intimidates. (60)

- * (13) Identify and explain the hardware and software characteristics of the three types of computer systems (mainframe, mini, and micro) and describe the capabilities/limitations and advantages/disadvantages of each. (1,27,29,57,63,65,66,67,31,71,81,84,78)
- * (3) Explain (compare and contrast) the concepts of networking and time sharing between mainframe computers and microcomputers. (65,81,2)
- * Define "intelligent" terminals and give examples of their use.
- * Distinguish between dedicated networks and switched/dialed networks.
- * Distinguish between parallel and serial communication.
 - (1) Discuss the advantages and disadvantages of interactive versus batch computer systems. (86)
- * (5) Define and be able to distinguish between different categories of computer software: operating system, application, compiler, etc. (2,30,72,81,86)
 - (3) Understand the capabilities and limitations of computer software. (51,81,79)
- * (11) Identify and be able to distinguish between major hardware and software characteristics. (56,46,67,1,2,20,25,30,66,70,82)
 - (8) Evaluate and select computer systems (hardware and software) that are appropriate for specific functions and needs. (64,15,6,37,55,42,46,78,52)
- * (11) Demonstrate proper care and maintenance of hardware and software devices. (11,28,51,52,9,15,48,64,69,78,81)
- * (19) Trace the historical development of computers and be able to discuss the resulting societal effects of the past, present and future. (17,24,41,51,52,54,69,71,78,81,82,20,27,23,35,38,55,63,87)
- * (8) Identify and explain the events leading to the 4 main stages of computer development: vacuum tubes, transistors, integrated circuits, microprocessors. (23,35,38,55,63,81,82,87)
- * (3) Identify several early computing devices and compare each to modern computers. (17,81,82)
 - (2) Trace the history of the computer from Babbage to present day. (27,81)
 - (1) Discuss the historical significance of the Eckert/Mauchly-Atanasoff and von Neumann controversies. (68)
 - (1) Explain the evolution of computer languages.
- * (1) Understand the rapid pace of change in the development of computers since 1940's. (20)

- * (6) Operate a variety of computer systems (including configuration and connection) and generalize common characteristics. (52,71,81,9,69,78)
- * (19) Operate a microcomputer: on-off sequences, loading, running, saving, copying, and printing. (17,22,26,30,31,33,36,41,48,49,51,58,69,50,78,79,80,81,85)
 - (4) Demonstrate keyboarding skills. (4,51,78,81)
 - (1) Read and understand a computer manual provided with a home computer. (36)
 - (1) Use the same application on two different computer systems to demonstrate knowledge of similarities and differences between systems. (13)
 - (1) Operate computer system without referring to manuals. (19)
 - (1) Follow directions in a menu-driven program. (81)
 - (1) Demonstrate use of system commands such as CATALOG, LOAD, RUN, LIST, SAVE, RENAME, DELETE, LOCK, UNLOCK. (81)
 - (1) Operate computer systems with more sophisticated peripheral equipment. (81)
- * Identify how a person can access a computer; e.g.,
 1. via a keyboard terminal
 - a. at site of computer
 - b. at any distance via telephone lines
 2. via punched or marked cards
 3. via other magnetic media
- * Identify the need for data to be organized if it is to be useful.
- * Identify the fact that information is data which has been given meaning.
- * Identify the fact that data is a coded mechanism for communication.
- * Identify the fact that communication is the transmission of information via coded messages.

COMPUTER APPLICATIONS:

- * (1) Computer applications refer to the ways in which computers are employed to do useful work. These applications include: artificial intelligence, CAD/CAM, CAI/CMI, communication, data base management, financial management, simulation, sound, word processing. (74)
- * Identify the following as types of computer applications:
 - a. information storage and retrieval-record keeping, data base management
 - 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
 - j. computer-assisted instruction
 - k. computer-managed instruction
 - l. problem solving
 - m. forecasting
 - n. market research
 - o. game playing
- *(27) Describe, give examples, and identify advantages and disadvantages of computer software applications (listed above) that are encountered in certain areas: homes, recreation, business and industry, medicine, law and enforcement, transportation, military defense, weather prediction, research and education, libraries, government, creative arts. (11,12,13,66,23,58,64,25,79,69,81,45,36,57,41,38,67,3,21,40,71,22,43,52,82,84,87)
- (4) Describe in detail at least one specific application of value in user's own career area. (29,30,36,65)
- (3) Have used a variety of applications (e.g., database, word processing, canned programs). (7,24,65)
- * (1) Recognize the definition of and some advantages of computer modeling/simulation (e.g., forecasting). (81)
- (2) Explain the objectives of and use of a popular microcomputer software package (e.g., visicalc, dBASE II). (66,81)
- * Determine how computers and computer-supported applications can assist an individual as he/she plays various roles i.e., consumer, worker, citizen; and how such systems assist groups and organizations as they attempt to accomplish tasks and responsibilities.
- * Design, develop, and implement a computer-supported application that would be personally useful.
- (1) Distinguish between the applications of general purpose vs. special purpose computers. (68)
- (1) Train others in the use of a particular application program or package. (16)
- (1) Aim for actual use of a few applications, rather than a smattering of knowledge about many. (9)

- (1) Describe two similarities and two differences between a robot's application and a business application using a desktop computer. (45)
- *(3) Explain that computer functions are generally useful for effective information-processing tasks that require any of the following: (69,79,81)
 - a. handling large amounts of information (searching, sorting, deleting, updating, summarizing)
 - b. rapid handling of information
 - c. accuracy
 - d. repetition
 - e. the storage and retrieval of information in an accessible form
- *(7) Determine when a computer should be used and be able to identify some of the factors to consider when making decisions regarding the use of computers (13,29,58,69,70,78,81):
 - a. cost
 - b. people's attitudes (fears, anxieties)
 - c. availability of suitable hardware and software/applications
 - d. hardware limitations-memory capacity, lack of peripherals, etc.
 - e. complexity of some computer-supported applications
 - f. application appropriateness
 - g. ethical and moral considerations
- *(10) Determine what hardware/software is necessary to solve particular problems or suit certain applications. (50,73,49,69,57,3,12,16,17,30)
- *(5) Describe, and indicate strengths and weaknesses of computer in such areas as problem solving, creativity, skill development. (27,13,40,12,78)
- *(13) Understand and explain the capabilities and limitations of computers. (17,20,22,31,36,52,50,66,72,81,82,79,87)
- *(2) Distinguish between human capabilities and computer capabilities. (69,81)
- (6) Explain that the operator controls the computer system rather than the system controlling the population and that computers do not make mistakes, people make the mistakes. (10,11,25,26,71,1)
- *(4) Realize computers are not an end in themselves, but a tool to be utilized in making our lives better through proper application. Realize that computers will not solve all our problems, rather will provide us with greater efficiency and access to more information to help us solve problems. (5,12,38,68)

- * Recognize that innovations in computer hardware and software continually expand the potential utility of the computer.
- *(12) Explain current and future applications of computers in relation to the impact on society. (41,51,71,72,11,20,40,69,81,82,79,86)
- * Recognize that for at least the short-term future, computers will continue to be made smaller while the amount of information they can hold will continue to increase.
- *(19) Understand and be able to discuss the legal and ethical computer related issues. (e.g., copyright law, computer crime, privacy, security) (51,1,7,34,82,78,47,44,2,4,14,31,8,34,38,45,53,55,20)
- *(23) Discuss some of the positive and negative consequences of computer use in today's society. (impact on society). (21,15,24,17,5,43,35,23,71,36,69,52,31,38,33,40,41,78,81,82,85,87)
- *(10) Identify computer-related occupations and jobs and explain in general the job description and training for each. (17,51,71,10,11,69,78,81,82,87)
- * Recognize that computerization both increases and decreases employment.
 - (1) Discuss three ways in which the computer has impacted the way people do their jobs in the banking industry. (86)
 - (1) Discuss what is meant by the "cashless society". (86)
 - (2) Name five areas in which computerized robots have replaced human labor. (example: canning factories, meat packing, auto manufacturing, etc.) Discuss the relationship between computerization and manual systems in their field. Discuss realistic probability of the computer replacing humans in certain endeavors. (44,25)
 - (1) Explain factors that make automation attractive - volume, routine, etc. (20)
 - (1) Do you feel our present day space program has helped develop our home computers? Explain. (32)
 - (1) Identify areas served better by computers than pre computer era. (24)
 - (1) Distinguish societal needs for supercomputers vs. computer on a chip. (68)
- * Recognize that computerization can personalize and impersonalize procedures in fields such as education.
 - (2) Demonstrate an awareness of computer-related technologies and new and developing industries to support systems. (11,22)

- (1) Demonstrate the ability to use mainframe computers. (65)
- (1) Understand computing for young people. (64)
- * (1) Use software documentation. (51)
- (1) Analyze the RS 232 requirements of 2 pieces of equipment and set them to be compatible. (73)
- (1) Seek out information from local, state or national data bases. (46)
- (1) Determine the strategies for 1) minimizing the probability of computer error 2) minimizing the consequences of computer error in a specific system (e.g., payroll, library). (43)
- (1) Use a computer to simulate causal loops using a simulation language such as Micro-dynamo. (37)
- (1) Understand computer use during leisure time. (55)
- (1) Describe a problem that is solved by a computer doing many iterations. (25)
- (1) Use the microcomputer to gather, organize, analyze, process, and evaluate information. (17)
- (1) Understand meaning of and give of GIGO. (44)
- (1) Generalize and interpret data. (22)
- (1) Identify ways to obtain information about computers when you don't have competency yet. (3)
- (1) Video games! (8)
- (1) Artificial intelligence. (8)
- * (4) List/identify sources of software and hardware information. (12,15,17,31)
- * (20) List evaluation criteria and evaluate software and hardware in terms of usefulness to particular application. (quality, content, use of graphics, etc.) (65,7,16,23,30,31,38,40,44,52,70,71,72,67,22,42,56,79,80,81)
- * (5) Apply their understanding of computer capabilities and applications to the selection of software that enables them to use the computer as a tool in managing information, processing text, and solving problems. (12,22,38,42,56)
- (1) Utilize a generally accepted evaluation scheme to write reviews of a number of educational software programs. (39)
- * (3) Define word processing and know the capabilities and limitations as well as the advantages and disadvantages of using word processing. (32,49,81)
- * (16) Create, edit, save, and print multi-page documents using word-processing software regularly for academic or personal uses. (78,2,12,26,33,39,46,48,54,62,63,65,73,9,37,80)

- (1) Use a word processing program to edit an existing document. (1)
- (2) Using a word processing software package, design a form letter and merge it with a list of names and addresses to create personalized letters. (47,X)
- * Describe the major differences between word processing equipment capabilities as relating to: display and output methods, storage and access methods, and text editing and formatting.
- * List the basic steps required to implement a word processing system.
- * Identify a number of commercial word processing software packages and compare capabilities.
- *(14) Utilize a data base management system (create, search, sort, retrieve, print, and edit). (1,2,78,37,39,44,47,48,61,65,62,12,80,81)
- *(1) Define data base management and explain the capabilities and limitations as well as the advantages and disadvantages of using file management software. (49)
 - (1) Retrieve requested data on existing data base. (1)
 - (1) Create a dictionary for a data base management system. (2)
- * Identify a variety of commercial data management/record keeping software packages and compare capabilities.
- * Identify some advantages and 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.
- *(11) Utilize a spreadsheet software system. (1,6,33,78,37,39,46,48,54,65,80)
 - * Define financial management software packages and list capabilities and limitations.
 - * Identify commercial financial management software packages and compare capabilities.
 - (1) Use Visicalc or another model (simulation). (6)
 - (1) Modify data contained in an existing electronic spreadsheet and produce new totals. (1)
 - (3) Use simple statistical programs. (48,63,73)
 - (1) Describe how a computer is used in testing and statistical treatment of data. (81)
 - (4) Create a computer driven graphic display. (2,26,62,88)
 - (1) Determine the most beneficial uses of graphics in program design and/or selection. (55)
 - (1) Demonstrate how computer graphics are used for visual display of graphs and illustrations. (81)

- (1) Write an instructional message involving both graphics and/or sound. (88)
- *(12) List, explain, and give examples of the major uses of microcomputers in education/instruction. (management of instruction-CMI, delivery of instruction-CAI, games, word processing, teacher utility, data management, etc.) (15,33,39,61,67,70,78,80,81,85,86,88)
- *(1) Critically discuss the philosophical and psychological basis for the LOGO language and its use with students at various age levels (capabilities and limitations). (80)
- *(4) Demonstrate the capabilities of using LOGO in educational situations. Develop age appropriate activities to conduct in learning LOGO consistent with the LOGO philosophy. (39,56,71,80)
- (4) Evaluate the effectiveness of instructional computer programs and courses that use computerized teaching materials (educational soundness, quality of program documentation, ease of use of the program, degree of student control, visual appearance and variety, ease of correcting input errors, option within program for assistance, presentation of feedback, cost, ease of making modifications). (15,2,78,85)
- (6) Design and/or select and appropriately use computer programs to enrich and extend the regular course instruction (manage and develop an environment in which computers are available as teaching/learning tools). (17,15,44,33,78,85)
- *(3) Explain in general the functioning of CMI systems. (15,78,85)
- (3) Explain the uses of simulations as teaching tools. (15,55,78)
- (1) Identify materials and projects related to computer education: professional journals, sources of software, educational research and development centers. (85)
- (1) Name two computer languages well suited to educational applications. (81)
- (2) Distinguish between learning about and learning with computers and indicate where the roles of both could be used effectively. (44,80)
- (1) Move beyond our current curricula to curricula that stress understanding and thinking skills with the use of data bases, while playing down the rote. (49)
- (2) Demonstrate awareness of dangerous side of computers in education as well as positive side. (potential and implications of computerization on education) (27,37)

- (2) Use instructional games appropriately and effectively in teaching. (15,78)
- (1) Demonstrate knowledge and value of the processes of involving students in instructional materials development. (15)
- (2) Improve less than adequate instructional computer programs. (15,17)
- * (4) Integrate CAI into curriculum subject areas. (37,58,78,85)
- * (3) Identify and explain the characteristics of the variety of CAI software types: simulation, tutorial, etc. (64,35,85)
- (3) Demonstrate a familiarity with computer simulations and models. (15,55,78)
- (4) Evaluate various computer assisted instruction packages. (2,15,78,85)
- * (3) Identify CAI authoring languages such as PILOT and be able to use it to write CAI lessons. (69,78,81)
- (2) Explain the similarities and difference between CAI and CMI. (71,85)
- (2) Relate principles of traditional learning theories to CAI material. (44,27)
- * Define and explain the purpose and advantages and disadvantages of authoring languages such as PILOT.
- (1) Explain the design of effective drill and practice materials. (15)
- (1) Apply computerized drill and practice in a variety of teaching situations. (15)
- (2) Select a CAI package that teaches problem solving skills. (6,78)
- (1) Explain the relationship between individual differences and computer assisted learning: understanding of how different learning styles and cognitive skills interact with computer enhanced learning. (85)

COMPUTER PROGRAMMING

- Computer programming refers to the instructions and commands that are needed to direct the operation of the computer. Topics may include: algorithms, logic, computer languages, operating systems, flowcharting, problem solving.
- * (3) Understand the language of software packages and operate software packages, including ability to run, review, and revise a simple program. (51,71,82)
 - (4) Identify problems which can be solved by computer and which cannot. (70,63,69,82)

- (3) Understand enough about programming to know the nature of good programs. (31,51,36)
- *(4) Develop user-oriented documentation for programs. (44,2,16,82)
 - (1) Understand the role of the programmer, instructional designer, and target audience in software/courseware selection. (19)
 - (1) Locate a programmer and describe a programming need to a programmer. (12)
- *(1) Understand what a good programmer can make a computer do. (49)
 - (1) Understand and be able to utilize the processes of involving students in instructional materials development. (15)
 - (3) Absolutely not necessary to be computer literate! An irrelevant skill for classroom teacher. (46,67,19)
- *(2) Define algorithm and flowchart. (25,81)
- *(9) Develop an algorithm and flowchart for the solution of a stated problem corresponding to their level of expertise in a given field of study. (22,20,24,57,3,69,81,68,86)
- *(14) Create a flowchart (structured pseudocode) to solve a simple problem. (69,2,5,23,49,72,63,32,44,39,3,81,82,86)
 - (3) Read an algorithm or flowchart and describe the output. (35,18,81)
- *(1) Develop a flowchart for a given algorithm. (81)
- *(2) Identify and explain the function of the symbols used to develop a flowchart. (81,82)
 - (1) Possess a repertoire of useful algorithms. (8)
- * Select an appropriate algorithm from a set of alternatives using criteria such as efficiency, elegance, and appropriateness.
 - (1) Explain why flowcharts are helpful in developing computer programs. (82)
 - (1) Eliminate the word (and concept of) flowcharting. (60)
- *(9) Identify high level programming languages and their capabilities and limitations in relation to their appropriateness for various applications and grade levels. (64,71,12,13,5,17,31,82,86)
- *(2) Understand the use of authoring languages as an alternative to generating software in BASIC etc. (64,85)
- *(2) Recognize that computers follow sequences of instructions written in a language with tightly defined rules of syntax and grammar understandable to both humans and computers. (29,81)

- *(1) Define and explain the capabilities and limitations of low-level languages. (8)
- *(1) Critically discuss the philosophical and psychological basis for the LOGO language and its use with students at various levels. Identify advantages and disadvantages of a language such as LOGO. (80)
- * Contrast high-level and low-level (assembly) languages.
- * State and define the two kinds of rules common to all programming languages.
- * Define machine language.
- * Explain the relationship between high level languages and machine language.
- * Define and explain the purpose and advantages and disadvantages of authoring languages such as PILOT.
- (3) Demonstrate a knowledge of structured/modular programming in any language. (69,70,38)
- *(7) Describe the steps in creating a computer program, from problem definition through to implementation. (66,65,50,36,82,21,81)
- (18) Write computer programs in BASIC. (65,71,66,58,56,82,54,57,37,17,33,73,26,69,78,80,84,81)
- *(25) Write simple reasonable programs in one or more high-level language. (61,39,54,50,52,48,66,32,43,12,30,9,21,52,69,22,3,6,23,81,68,44,1,26,86,88)
- *(8) Write introductory programs in LOGO. (17,33,37,69,73,78,80,81)
- *(3) Write instructional programs in authoring language such as PILOT. (69,78,81)
- (7) Write computer programs in PASCAL. (33,73,66,69,78,84,81)
- *(12) Modify computer programs written by others. (15,17,29,30,55,57,72,9,21,52,69,78)
- *(9) Given a program containing errors (syntax and logic), the student will debug the program so it will execute properly. (1,2,26,43,55,61,58,78,82)
- *(6) Read a simple program, explain the statements and logic of the program and describe the output. (2,3,40,55,57,78)
- *(4) Translate a given algorithm or flowchart into a high-level programming language. (22,1,26,69)
- (3) Use top-down design method for studying programming strategies. (38,69,87)
- (2) Write a program using a loop to solve a problem. (54,88)
- (2) Identify simple programming commands. (51,11)

- (1) Differentiate between artificial intelligence and traditional logic patterns in programming. (45)
- (1) Discuss the arguments presented by both sides in the controversy of the instructional aspects of teaching BASIC vs. a structured programming language. (80)
- (1) Write a program in either BASIC, pascal, or FORTRAN to accomplish a specific task. (66)
- (1) Program a simple algorithm without reference to book/manual. (63)
- (2) Demonstrate preliminary skill in the design and construction of a simulation. (15,55)
- (2) Record (save) a program on a diskette. (12,88)
- * Explain how a program is run in a computer.
- (1) Intermediate and advanced programming is necessary solely for students demonstrating potential for success in field. (11)
- (1) Write a pseudo code solution for a given problem. (35)
- (2) Develop programs to perform tasks in areas such as social studies, business, science, and mathematics. (69,78)
- (1) Read a Pascal/basic program that is a simulation on the order of Lemonade for the Apple microcomputer. (6)
- (1) Direct computer to perform some simple function (e.g., output name, count to ten, etc.) (41)
- * Define the following: variable, string variable, linear programming, branched programming, bug, debugging, coding, statement, syntax, semantics, data names, looping, unconditional branch, conditional branch, end of file.
- * (10) Analyze problems and select and/or develop appropriate problem solving strategies for their solution. (69,64,41,25,7,24,13,78,81,87)
- (3) Recognize that programming is a problem-solving process. (21,81,87)
- (1) Understand that human problem solving involves tacit knowledge or tasks and computer solution of similar problems does not (e.g., in summing numbers, computers must initialize the sum, humans don't). (13)
- (1) Application of high level thinking skills called into play in programming and other problem solving situations: "meta-awareness". (4)
- (1) Ability to translate algebra story problem into an equation. (18)
- (1) Understand computer logic. (65)
- (1) Analyze tasks to determine appropriateness of use of computer for problem solving. (81)

COMPUTER ATTITUDES

- (3) Demonstrate positive attitudes toward computers. (65,33,84)
- *(2) Feel confident in their ability to select appropriate hardware and software to use the computer as a tool in the home, school, and work environment. (7,22)
- (3) Feel comfortable with the use of computers. (31,50,79)
- (7) Accept computer as another fast and accurate problem solving tool. Given opportunity, use computer to solve problems. View the computer as a helper rather than a threat. (19,21,30,52,65,57,27)
- *(10) Given a choice of activities, including free time, is willing to use a computer for tasks associated with his/her school, work or personal life. (26,30,70,61,56,29,3,48,54,88)
- *(2) Demonstrate an appreciation for the potential threats to personal privacy that computers pose. (47,68)
- *(4) Be free of fear and intimidation when using a computer. (11,17,21,29)
- *(15) Demonstrate an attitude of computer professionalism or ethics: issues dealing with misuse of computers such as stealing computer programs, breaking security codes, borrowing code without acknowledgement. (17,34,47,44,2,4,14,31,34,38,45,47,53,55,71)
- (3) Realize and explain that "computer mistakes" are people mistakes. Recognize that machines make errors when humans fail to provide proper coding. (1,26,86)
- (2) Own a computer or demonstrate a desire to own a computer. (70,88)
- (1) Demonstrate an open mind towards computers. (46)
- (1) Demonstrate respect for computer. (79)
- (1) Write a paper debunking the viewpoints of popular literature concerning computers (e.g. choose a novel or film which depicts a computer in a negative way and show the fallacy of this view). (66)
- (1) Volunteer positive comments about the impact of computers on their ability to teach effectively in the future. (56)
- (1) Spontaneously suggest situations in which the computer can be used. (57)
- (1) Define and defend a specific application of computer technology. (36)
- (1) When given a choice of media, machine, and/or institutions, identifies the computer as an entity that will bring about positive changes in society. (54)

- (1) Demonstrate an awareness of the value of involving students in the development of computerized instructional materials. (15)
- (2) Realize and explain that computers do not control people and feel that they as people control computers. (50,59)
- (1) Gets angry with people, not machines. (26)
- (1) Demonstrate a personal desire to improve and the realization of the concept of "life-long learning". (46)
- (1) Discuss realistic probability of the computer replacing humans in certain endeavors. (44)
- (1) Identify the reasons people have anxiety towards computers. (23)
- (1) Understand how computers are appropriately part of their leisure time (games, entertainment). (55)
- (1) Demonstrate an attitude of responsibility for documenting programs. (2)
- (1) Develop the attitude that if we as earthlings are to communicate and utilize all the information that will be generated, being computer literate is a must.(5)
- (1) Does not perpetuate computer myths. (12)
- * Enjoy and desire to work or play with computers, especially computer-assisted learning.
- * Describe past experiences with computers with positive-affect words like fun, exciting, challenging.
- * Value the potential role of computers in meeting societal and institutional needs.
- * Value efficient information processing provided that it does not neglect accuracy, the protection of individual rights, and social needs.
- * Value computerization of routine tasks so long as it frees people to engage in other activities and is not done as an end in itself.
- * Value 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.
- * Appreciate the proper utilization of computer technology as it assists people without hurting society and the environment.

COUNTER-ATTITUDES

- (2) Anxiety toward computer may be beneficial.
- (6) Computer attitudes is not part of computer literacy. (73,63,43,73,9,8)

- (9) Computer attitudes do not have to be positive. (e.g., "I worry about the good of having "positive" attitudes toward computers. I'd opt for restatement of "neutral" attitudes or "informed", or "untainted" or something a bit less jingoistic than positive." "Do all people really have to have positive attitudes towards computers? Can't they be ambivalent or concerned?" "Why do attitudes have to be positive?" "Realistic attitudes, not necessarily positive.") (60,64,52,53,13,8,73,83,86)

OTHER

- (1) Computer impact issues, problems involving the broader effects of computers on society (including ethics), belong in a separate section and are vitally important to study in and of themselves. (83)
- (1) Delete most of the competencies in the "computer systems area such as historical and terminological knowledge which I don't personally consider terribly important. (75)
- (1) I am disappointed in the lack of content in your four areas. (73)
- (1) Robotics-big track, turtle, TOPO, etc. (64)
- (1) Computer literacy should be taught in an integrated manner. Application of computers should be used to link the other components. (38)
- (1) "Competencies" is misleading. (31)
- (1) "Computer Literate" does not mean in-depth knowledge of computers and their applications, but does mean the knowledge and understanding to be able to use computers and make intelligent decisions about their use. (31)
- (1) The flavor of the definitions you give here is awfully academic - knowledge about rather than experience with. (9)

APPENDIX C:
DOMAIN SPECIFICATIONS
(WITH FIRST SELECTION OF COMPETENCIES)

TEST SPECIFICATIONS FOR A CRITERION-REFERENCED
COMPUTER LITERACY ASSESSMENT INSTRUMENT

These test specifications will be used to guide a committee of test-item writers in the development of test items for a criterion-referenced computer literacy assessment instrument. The test specifications will provide an unambiguous definition of the domain of computer literacy competencies to be measured, and will specify a set of rules for generating a sample of test items from the domain. Adherence to clearly defined test specifications during the development of test items is necessary to establish the content validity of the test.

PURPOSE OF THE TEST

The computer literacy assessment instrument will attempt to measure the computer literacy knowledge and skills of undergraduate students at Iowa State University prior to taking college-level computer literacy courses. The results of the test may be used in the following ways:

1. To document the pre-college computing competencies of students at Iowa State University and develop normative data about various subgroups of students.
2. To justify revisions in the content of introductory computing courses to reflect student needs and prerequisite skills.
3. To provide criteria for the placement of students in appropriate computing courses based on their level of computer literacy.

TEST CHARACTERISTICS

The test of computer literacy will consist of four subtests based on the four part computer literacy definition and corresponding competencies (see DOMAIN SPECIFICATIONS). The total number of test items on the test has not been determined yet because the simplicity or complexity of the test items will influence the number of items that can be completed in a one hour test. It is estimated that the test will contain 50 to 100 items of varying complexity.

The competencies are written in terms of the knowledge, skills, and attitudes the examinees should be able to demonstrate by selecting the correct responses to multiple choice or Likert-type items. While many of the competencies appear to be expressing knowledge and comprehension level competencies, the test items themselves will attempt to measure higher level cognitive skills. According to the Educational Testing Service, "A well conceived and constructed multiple-choice question should require a student to select, weigh, and apply what he knows in order to answer the questions correctly" (ETS, 1973).

DOMAIN SPECIFICATIONS

COMPUTER LITERACY DEFINITION

Computer literacy is operationally defined as an understanding of computer characteristics, capabilities, and applications, as well as the ability to implement this knowledge in the skillful, productive use of computer applications suitable to individual roles in society. The knowledge and skills of computer literacy as defined above have been divided into four major categories; computer systems, computer applications, computer programming, and computer attitudes. These four sections are further defined in the following list of computer literacy competencies.

COMPUTER LITERACY COMPETENCIES

A. COMPUTER SYSTEMS

Computer systems refer to the appropriate, knowledgeable use of equipment (hardware) and programs (software) necessary for computer applications. This requires understanding and abilities in the following areas: computer functions, computer hardware, computer software, computer systems configuration, computer terminology, historical development, and the operation of computers.

GENERAL COMPUTER FUNCTIONS

1. Demonstrate an understanding of the binary system and its functional relationship to digital computers.
2. Demonstrate an understanding of the process of data processing in a computer (the transformation of data by means of a set of pre-defined rules and computers process data by searching, sorting, deleting, updating, summarizing, and moving.)
3. Demonstrate an understanding of the primary function of a computer system as the input of information, processing of information, and output of information.
4. Demonstrate an understanding of the idea that a computer is an impersonal, literal machine (hardware) incapable of functioning without a set or program of instructions written in a programming language (software).
5. Demonstrate an understanding of the function of an operating system and be able to compare and contrast the characteristics of different operating systems (e.g., DOS, CP/M).
6. Identify and distinguish between analog and digital computing operations.

COMPUTER HARDWARE

7. Demonstrate an understanding of the capabilities and limitations of computer hardware.
8. Demonstrate an understanding of the characteristics (size, functions, limitations and differences) of RAM, ROM, EPROM, floppy disc memory, hard disc memory, bubble memory and other types of memory.
9. Understand and use computer terminology: hardware, software, CPU, memory, input, output, network, compiler, interpreter, machine language, bit, byte, kilobyte, K, on-line, off-line, NIBBLE, hexa-decimal, chip, mainframe, microprocessor, DOS, binary, time-share, RAM, ROM, BASIC, vacuum tube, transistor, integrated circuit, I/O, bug, peripheral, disk, disk drive, printer, keyboard, microprocessor, program, hardcopy, CRT, CAI, CMI, modulator etc.
10. Identify and demonstrate an understanding of the function of the major components of a computer system: input, output, CPU (control unit, arithmetic/logic unit), secondary storage, and memory.
11. Identify and demonstrate an understanding of the purpose/function of the control unit and the arithmetic/logic unit of the CPU of a microprocessor.
12. Identify and distinguish between special purpose and general purpose computers.

COMPUTER SOFTWARE

13. Define and distinguish between different categories of computer software: system software and application software.
14. Demonstrate an understanding of the capabilities and limitations of computer software.

COMPUTER SYSTEMS CONFIGURATION

15. Identify and demonstrate an understanding of the hardware and software characteristics of the three types of computer systems (mainframe, mini, and micro) and describe the capabilities/limitations and advantages/disadvantages of each.
16. Demonstrate an understanding of (compare and contrast) the concepts of networking and time sharing between mainframe computers and microcomputers.
17. Distinguish between parallel and serial communication.
18. Demonstrate an understanding of the advantages and disadvantages of interactive versus batch computer systems.

HISTORICAL DEVELOPMENT

19. Trace the historical development of computers and be able to discuss the resulting societal effects of the past, present and future.
20. Identify and demonstrate an understanding of the events leading to the 4 main stages of computer development: vacuum tubes, transistors, integrated circuits, microprocessors.
21. Identify several early computing devices and compare each to modern computers.

THE OPERATION OF COMPUTERS

22. Demonstrate the ability to operate a variety of computer systems (including configuration and connection) and generalize common characteristics.
23. Demonstrate the ability to operate a microcomputer: on-off sequences, loading, running, saving, copying, and printing. Demonstrate the appropriate use of systems commands such as CATALOG, LOAD, RUN, LIST, SAVE, RENAME, DELETE, LOCK, UNLOCK.
24. Demonstrate an understanding of common hardware failures and the proper care and maintenance of hardware and software devices.

B. COMPUTER APPLICATIONS

Computer applications refer to the ability to responsibly evaluate, select, and implement a variety of practical computer applications to do meaningful and efficient work based on an understanding of the following: general types of applications, capabilities and limitations of applications, societal impact (past, present, and future), evaluation and selection techniques, and specific applications (word processing, data base management, spreadsheet/financial management, statistical analysis, graphics, and educational applications).

GENERAL TYPES OF APPLICATIONS

1. Identify and give examples of the following as types of computer applications:
 - a. information storage and retrieval-record keeping, data base management
 - 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
 - j. computer-assisted instruction
 - k. computer-managed instruction
 - l. problem solving
 - m. forecasting
 - n. market research
 - o. game playing
2. Demonstrate an understanding of, (identify advantages and disadvantages) of computer software applications (listed above) that are encountered in certain areas: homes, recreation, business and industry, medicine, law and enforcement, transportation, military defense, weather prediction, research and education, libraries, government, creative arts.

CAPABILITIES AND LIMITATIONS

3. Demonstrate an understanding of the idea that computer functions are generally useful for effective information-processing tasks that require any of the following:
 - a. handling large amounts of information (searching, sorting, deleting, updating, summarizing)
 - b. rapid handling of information
 - c. accuracy

- d. repetition
 - e. the storage and retrieval of information in an accessible form
4. Determine when a computer should be used and be able to identify some of the factors to consider when making decisions regarding the use of computers:
 - a. cost
 - b. people's attitudes (fears, anxieties)
 - c. availability of suitable hardware and software/applications
 - d. hardware limitations-memory capacity, lack of peripherals, etc.
 - e. complexity of some computer-supported applications
 - f. application appropriateness
 - g. ethical and moral considerations
 5. Determine what hardware/software is necessary to solve particular problems or suit certain applications.
 6. Demonstrate an understanding of, and indicate strengths and weaknesses of computers in such areas as problem solving, creativity, skill development.
 7. Demonstrate an understanding of the idea that an operator controls the computer system rather than the system controlling the population and that computers do not make mistakes, people make the mistakes.
 8. Demonstrate an understanding of the idea that computers are not an end in themselves, but a tool to be utilized in making our lives better through proper application by providing us with greater efficiency and access to more information to help us solve problems.

SOCIETAL IMPACT

9. Demonstrate an understanding of past, present, and future applications and implications of computers in relation to the impact on society.
10. Demonstrate an understanding of legal and ethical computer related issues. (e.g., copyright law, computer crime, privacy, security) and make responsible decisions based on this understanding.
11. Identify computer-related occupations and jobs and demonstrate an understanding of the job description and training for each.

EVALUATION AND SELECTION

12. Identify sources of software and hardware application information.
13. Identify evaluation criteria, and demonstrate the ability to evaluate and select computer software and hardware in terms of usefulness to particular application or problem.

WORD PROCESSING

14. Define word processing and understand the capabilities and limitations as well as the advantages and disadvantages of using word processing.
15. Demonstrate an understanding of the utilization of word processing software programs to create, edit, save, and print multi-page documents regularly for academic or personal uses.

DATA BASE

16. Demonstrate an understanding of the utilization of a data base management system. (create, search, sort, retrieve, print, and edit)
17. Define data base management and understand the capabilities and limitations as well as the advantages and disadvantages of using file management software.

SPREADSHEET/FINANCIAL MANAGEMENT

18. Define and demonstrate an understanding of the utilization of a spreadsheet software system.

STATISTICAL

19. Identify and demonstrate an understanding of the utilization of simple statistical programs.

GRAPHICS

20. Demonstrate an understanding of procedures to create computer driven graphic displays.

EDUCATIONAL APPLICATIONS

21. Demonstrate an understanding of, and identify examples of the major uses of microcomputers in education/instruction. (management of instruction-CMI, delivery of instruction-CAI, games, word processing, teacher utility, data management, etc.)
22. Demonstrate an understanding of the capabilities of using LOGO in educational situations and the development of age appropriate activities consistent with the LOGO philosophy.
23. Demonstrate the ability to evaluate the effectiveness of instructional computer programs and courses that use computerized teaching materials.
24. Demonstrate the ability to design and/or select and appropriately use computer programs to enrich and extend the regular course instruction (manage and develop an environment in which computers are available as teaching/learning tools).

25. Demonstrate an understanding of the functioning of CMI systems.
26. Demonstrate an understanding of the uses of simulations as teaching tools.
27. Identify two computer languages well suited to educational applications.
28. Distinguish between learning about and learning with computers and indicate where the roles of both could be used effectively.

CAI

29. Demonstrate the ability to integrate CAI into curriculum subject areas.
30. Identify and demonstrate an understanding of the characteristics of the variety of CAI software types: simulation, tutorial, etc.
31. Demonstrate the ability to evaluate various computer assisted instruction packages.
32. Identify CAI authoring languages such as PILOT and demonstrate the ability to use it to write CAI lessons.
33. Demonstrate an understanding of the relationship between individual differences and computer assisted learning: understand how different learning styles and cognitive skills interact with computer enhanced learning.

C. COMPUTER PROGRAMMING

Computer programming refers to the ability to direct the operation of the computer through the skilled use of programming languages (high-level as well as software languages). This requires an understanding of problem solving strategies, algorithms and flowcharts, languages, programming skills.

PROBLEM SOLVING STRATEGIES

1. Demonstrate the ability to analyze problems and select and/or develop appropriate problem solving strategies for their solution.
2. Demonstrate an understanding that programming is a problem-solving process.

ALGORITHMS/FLOWCHARTS

3. Define and demonstrate an understanding of algorithms and flowcharts.
4. Demonstrate the ability to develop algorithms and flowcharts for the solutions to problems in individual fields of study.
5. Read an algorithm or flowchart and identify the output.
6. Identify and demonstrate an understanding of the function of the symbols used to develop a flowchart.
7. Demonstrate an understanding of why flowcharts are helpful in developing computer programs.

LANGUAGES

8. Identify high level programming languages and demonstrate an understanding of their capabilities and limitations in relation to their appropriateness for various applications and grade levels. (e.g., BASIC, PASCAL, COBAL, FORTRAN, LOGO)
9. Demonstrate an understanding of the use of authoring languages as an alternative to generating software in BASIC etc.
10. Demonstrate an understanding that computers follow sequences of instructions written in a language with tightly defined rules of syntax and grammar understandable to both humans and computers (understand the characteristics of and the relationship between the different types of languages: machine, assembly, high level, compiler, interpreter).

PROGRAMMING

11. Demonstrate an understanding of the utilization of software package languages to run, review, and revise programs.
12. Demonstrate an understanding of the characteristics of good programs.

13. Demonstrate the ability to develop user-oriented documentation for programs.
14. Demonstrate a knowledge of structured/modular programming in any language.
15. Demonstrate an understanding of the steps in creating a computer program, from problem definition through to implementation.
16. Demonstrate the ability to write simple reasonable programs in one or more high-level language. (e.g., BASIC, PASCAL, LOGO)
17. Modify computer programs written by others.
18. Given a program (or flowchart) containing errors (syntax and logic), identify the error and debug the program so it will execute properly.
19. Read a simple program, demonstrate an understanding of the statements and logic of the program, and identify the output.
20. Demonstrate the ability to translate a given algorithm or flowchart into a high-level programming language.
21. Define the following: variable, string variable, linear programming, branched programming, bug, debugging, coding, statement, syntax, looping, unconditional branch, conditional branch, pseudocode, recursion.

D. COMPUTER ATTITUDES

Computer attitudes refer to an individual's feelings about the personal and societal use of computers in appropriate ways. Positive attitudes include: an anxiety-free willingness or desire to use the computer, confidence in one's abilities to use the computer, and computer responsibility.

1. Demonstrate positive attitudes toward computers.
2. Feel confident in the ability to select appropriate hardware and software to use the computer as a tool in the home, school, and work environment.
3. Feel comfortable with the use of computers.
4. Accept the computer a fast and accurate problem solving tool. Given opportunity, use computer to solve problems. View the computer as a helper rather than a threat.
5. Given a choice of activities, including free time, is willing to use a computer for tasks associated with his/her school, work, or personal life.
6. Demonstrate an appreciation for the potential threats to personal privacy that computers pose.
7. Be free of fear and intimidation when using a computer.
8. Demonstrate an attitude of computer responsibility (ethics): issues dealing with misuse of computers such as stealing computer programs, breaking security codes, borrowing code without acknowledgement.
9. Demonstrate the attitude that computers do not make mistakes, rather computer operators are responsible for instructing the computer.

MULTIPLE CHOICE ITEM CHARACTERISTICS

The multiple choice items for this test will each consist of a stem, designed to present the problem or question, and a set of five response options (one correct option called the key and four incorrect options called distractors). The multiple choice items will be written with the following guidelines (Rahmlow and Woodley, 1979; Ward, 1981; Ebel, 1972):

ITEMS IN GENERAL

1. The ability measured by the test item should actually reflect the principal performance specified in the competency.
2. The concrete problem of selecting the correct answer must be an accurate reflection of the basic problem or competency the item was intended to present.
3. All items should be as simple as possible and still assess the competency as accurately as possible.
4. All items in the test should be independent. That is, no one item in the test should provide the correct answer to another item; and the correct answer to one item should not depend upon the ability to correctly answer another item.
5. The items for the computer literacy test must not be specific to any particular programming language, brand of computer, or commercial software package.
6. The item should be written so that the stem and all response options contribute to differentiating between competent and less than competent students.
 - a. Novel questions and novel problem situations can contribute to differentiating the critically minded examinees from the other examinees.
 - b. Emphasize items that demand more from the examinees than rote recall. The items should require the examinee to select, weigh, and apply knowledge in order to answer the question.
 - c. Diagrams, charts, graphs, flowcharts, programs, etc. should be used as often as possible if appropriate.

THE STEM

7. The stem must be an accurate statement of fact, presented as a specific question or incomplete statement, with one best correct answer.
8. The stem must be clearly and concisely written, and the problem specified must be easy to comprehend.
9. Include in the stem, any words that otherwise must be repeated in each response option.

THE RESPONSE OPTIONS

10. All of the options should be similar in all respects except the crucial one involved in the problem.
 - a. All options should be similar in grammatical structure.
 - b. All options should be similar in complexity, precision, and amount of clarification.
 - c. The options should be similar in length.
11. Avoid the use of "all of the above" or "none of the above" as response options.
12. All options should make sense in terms of the stem; that is, they should fit the stem.
 - a. All options should be meaningful. None should be absurd or nonfunctioning.
 - b. No two options should say the same thing.
 - c. Options should be worded believably; avoid the use of specific determiners or absolutes. (e.g., always, only, never, none, every, all)

THE KEY

13. The key should be a precise and correct answer to the question asked or completion of the incomplete statement.
14. Randomly assign the position of the key.
15. The key should not bear a unique association with the stem. (e.g., repetitious terms, synonyms)

THE DISTRACTORS

16. The distractors must be definitely incorrect, or clearly less appropriate than the key.
17. The distractors should not have anything obvious in common that is not shared by the key.
18. The distractors should meet the superficial and obvious parts of the specifications of the stem.
19. The distractors should be plausible and attractive to a person who does not know the correct answer.

FORMAT FOR INITIAL DRAFTS OF ITEMS

The initial drafts of the test items should be written on individual pieces of paper in the following format.

Competency: (e.g., A3 refers to competency #3 of the Computer Systems (A) section.)

Visual (if necessary): Include a rough sketch of the visual proposed for the item.

Stem:

Response Options:

- a.
- b.
- c.
- d.
- e.

The Key: Indicate the key with a star.

Test writer's name:

SAMPLE ITEMS

Competency: A20

Visual: none

Stem:

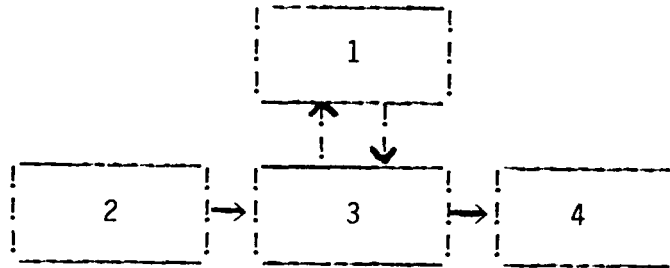
Four generations of modern computers can be characterized by which of the following sets of words? (the words must be in the correct chronological order.)

Response options:

- a. punched cards, floppy disks, hard disks, cassettes
- b. abacus, difference engine, analytical engine, electromechanical
- * c. vacuum tubes, transistors, integrated circuits, microprocessors
- d. relays, electromechanical, transistors, microprocessors
- e. mainframes, mini-computers, microcomputers, calculators

Competency: A10, A3, A9

Visual:



Stem: This is a schematic of a computer system with the arrows indicating the sequence of data movement within the system. Which of the following are the correct labels for each component?

Response options:

- * a. 1-memory
2-central processing unit
3-input device
4-output device
- b. 1-input/output device
2-secondary storage unit
3-central processing unit
4-random access memory
- c. 1-disk drive
2-keyboard
3-video display
4-printer
- d. 1-central processing unit
2-memory
3-input device
4-output device
- e. 1-arithmetic/logic unit
2-control unit
3-memory
4-input/output device

APPENDIX D:
JUDGEMENTAL ITEM ANALYSIS FORMS

ITEM CONTENT REVIEW

Reviewer: _____

Directions: Read each item and the competency it was intended to measure as indicated below. Indicate how well you feel each item reflects the objective it was written to measure. Judge a test item solely on the basis of the match between its content and the content defined by the objective (Is the content or skill measured by the test item a sample of the content or skill defined by the objective?)

Please use the five-point rating scale shown below to rate each item-objective pair as identified on this sheet. Please make comments or suggestions regarding the content of the items as you see necessary.

Poor Fair Good Very Good Excellent
1 2 3 4 5

Test A item #	Correct answer	Competency	Rating	comments
15	d	B27		
16	c	B26		
16	c	B30		
17	d	B30		
18	a	C19		
19	b	C12		
19	b	C14		
20	a	C19		
20	a	C21		
21	b	C19		
22	d	C21		
23	a	C12		
23	a	C13		

Test A item #	Correct answer	Competency	Rating	comments
24	b	C20		
25	c	C19		
26	b	C11		
27	d	B23		
27	d	B24		
28	b	B25		
29	c	B27		
29	c	B32		
29	c	C8		
30	a	B24		
30	a	B28		
31	b	B29		
31	b	B30		
32	a	B31		
33	e	B33		
34	d	A11		
35	a	A12		
36	a	A13		
36	a	A5		
37	e	A14		
38	b	A15		
39	c	A16		
40	b	A17		

Test A item #	Correct answer	Competency	Rating	comments
41	e	A18		
42	b	A19		
42	b	A21		
43	c	B2		
44	a	A15		
45	b	B4		
46	b	B6		
47	d	B8		
48	c	C3		
48	c	C4		
49	c	C5		
50	e	C7		
51	c	C9		
52	c	A1		
53	d	A3		
54	a	A22		
55	d	A23		
56	d	A24		
57	d	C21		
58	c	B4		
58	c	B13		
59	?	B15		
60	a	B16		

Test A item #	Correct answer	Competency	Rating	comments
60	a	B17		
61	c	B18		
62	a	B21		
62	a	B24		
62	a	B25		
63	b	B5		
63	b	B13		
64	c	A20		
65	c	A3		
65	c	C2		
66	c	A4		
67	a	A2		
68	a	A1		
69	d	A8		
69	d	A10		
70	c	B14		
71	e	C10		
72	c	C3		
73	a	C6		
74	b	B7		
75	a	A7		
75	a	A24		
76	c	A16		

Test A item #	Correct answer	Competency	Rating	comments
77	e	B1		
78	a	C8		
79	b	C21		
80	d	C15		
81	d	C13		
81	d	C21		
82	e	C8		
82	e	C10		
83	e	C8		
83	e	C10		
84	a	C8		
84	a	C10		
85	b	C8		
85	b	C10		
86	e	A10		
86	e	A11		
87	b	A19		
88	b	A7		
88	b	B3		
89	a	A6		
89	a	A9		
90	b	A9		
91	b	C1		

Test A item #	Correct answer	Competency	Rating	comments
91	b	C2		
92	d	B21		
93	b	B30		
94	a	B10		
95	b	B9		
96	b	A13		
97	e	B11		
98	b	B11		
99	e	B9		
99	e	B10		
100	b	A17		
101	c	B12		
102	c	B18		
104	b	A9		
105	b	B22		
106	c	A5		

TECHNICAL ITEM REVIEW

Reviewer: _____

Directions: Read each item and evaluate it according to the following questions. If you answer no to any of the questions for an item, please make a note of it directly on the test. Also mark any spelling, grammatical, punctuation, or typographical errors directly on the test. Finally, make a judgment as to whether you would accept the item for the final test with no revisions, accept the item with revisions, or reject the item completely (using the following three-point scale). Write the rating in the margin next to the item and circle it.

Reject	Accept w/revisions	Accept
1	2	3

1. Is the item clearly written for the intended group of examinees?
2. Is the item stem free of irrelevant material?
3. Is a problem clearly defined in the item stem?
4. Are the choices clearly written for the intended group of examinees?
5. Are the choices free of irrelevant material?
6. Is there a correct answer or a clearly best answer?
7. Have words like "always," "none," or "all" been removed?
8. Are likely examinee mistakes used to prepare incorrect answers?

9. Is "all of the above" avoided as a choice?
10. Are the choices arranged in a logical sequence (if one exists)?
11. Was the correct answer randomly positioned among the available choices?
12. Are all repetitious words or expressions removed from the choices and included in the stem?
13. Are all of the choices of approximately the same length?
14. Do the item stem and choices follow standard rules of punctuation and grammar?
15. Are all negatives underlined?
16. Are grammatical cues between the item stem and the choices, which might give the correct answer, removed?
17. Is the stem format appropriate for measuring the intended objective?

APPENDIX E:
REVISED COMPETENCIES
KEYED TO TEST ITEMS

COMPUTER LITERACY COMPETENCIESCOMPUTER LITERACY DEFINITION

Computer literacy is operationally defined as an understanding of computer characteristics, capabilities, and applications, as well as the ability to implement this knowledge in the skillful, productive use of computer applications suitable to individual roles in society. The knowledge and skills of computer literacy as defined above have been divided into four major categories; computer attitudes, computer systems, computer applications, and computer programming. These four sections are further defined in the following list of computer literacy competencies.

COMPUTER LITERACY COMPETENCIES KEYED TO THE FOUR MAJOR SECTIONS OF THE COMPUTER LITERACY ASSESSMENT INSTRUMENT

The specific test items that measure each of the competencies are listed in parentheses following each competency.

SECTION II: COMPUTER ATTITUDES

Computer attitudes refer to an individual's feelings about the personal and societal use of computers in appropriate ways. Positive attitudes include: an anxiety-free willingness or desire to use the computer, confidence in one's abilities to use the computer, and computer responsibility.

Note: The Computer Attitude competencies are keyed to the two parts of Section II rather than specific items. The items in Part One collectively measure an individual's degree of positive attitudes towards computers and the items in Part Two collectively measure an individual's degree of computer anxiety.

1. Demonstrate positive attitudes toward the appropriate use of computers. (Part One and Two)
2. Demonstrate a feeling of confidence in the use of computers as tools in the home, school, and work environment. (Part One and Two)
3. Demonstrate a feeling of comfort with the use of computers. (Part Two)

4. Demonstrate an attitude of acceptance of the computer as a fast and accurate problem solving tool (a helper rather than a threat). (Part One and Two)
5. Demonstrate a willingness to use a computer for tasks associated with school, work, or personal life. (Part One and Two)
6. Demonstrate an appreciation for the potential threats to personal privacy that computers pose. (Part One)
7. Demonstrate an attitude of responsibility for the ethical use of computers (Issues dealing with the misuse of computers; e.g., stealing computer programs, breaking security codes, borrowing code without acknowledgement are considered unethical). (Part One)
8. Demonstrate an attitude that computers usually are not responsible for "computer errors", and that computer users are responsible for instructing the computer. (Part One)
9. Be free of fear and intimidation when using a computer. (Part Two)

SECTION III: COMPUTER SYSTEMS

Computer systems refer to the appropriate, knowledgeable use of equipment (hardware) and programs (software) necessary for computer applications. This requires understanding and abilities in the following areas: computer functions, computer hardware, computer software, computer systems configuration, computer terminology, historical development, and the operation of computers.

GENERAL COMPUTER FUNCTIONS

1. Define the binary system and demonstrate an understanding of its functional relationship to digital computers. (1)
2. Demonstrate an understanding of the process of data processing in a computer (the transformation of data by means of a set of pre-defined rules and computers process data by searching, sorting, deleting, updating, summarizing, and moving). (2)
3. Demonstrate an understanding of the primary functions of a computer system (the input of information, processing of information, and output of information). (3)
4. Demonstrate an understanding of the idea that a computer is an impersonal, literal machine (hardware) incapable of functioning without a program of instructions (software). (4, 5)
5. Identify and distinguish between analog and digital computing operations. (6)
6. Identify and distinguish between interactive versus batch processing. (7)

COMPUTER SYSTEM CONFIGURATION

7. Define and be able to use computer system terminology (e.g., hardware, software, CPU, memory, input, output, network, bit, byte, kilobyte, K, on-line, off-line, hexa-decimal, chip, mainframe, microprocessor, DOS, binary, time-share, RAM, ROM, BASIC, vacuum tube, transistor, integrated circuit, I/O, bug, peripheral, disk, disk drive, printer, keyboard, microprocessor, program, hardcopy, CRT, modulator etc.). (8)
8. Identify and distinguish between special purpose and general purpose computers. (9)

9. Identify and demonstrate an understanding of the distinctions between the three types of computer systems (mainframe, mini, and micro). (10, 11)
10. Demonstrate an understanding of the concepts of networking and time-sharing computer systems. (12)
11. Identify and distinguish between parallel and serial communication. (13)
12. Demonstrate an understanding of the function of an operating system and be able to distinguish between the characteristics of different operating systems (e.g., DOS, CP/M). (14)
13. Demonstrate an understanding of common hardware and software failures. (15)
14. Demonstrate an understanding of the proper care and maintenance of hardware and software devices. (16)

COMPUTER HARDWARE

15. Demonstrate an understanding of the capabilities and limitations of computer hardware. (17)
16. Identify and demonstrate an understanding of the function of the major components of a computer system: input, output, central processing unit (control unit, arithmetic/logic unit), secondary storage, and memory. (18, 19)
17. Define and distinguish between the characteristics of RAM, ROM, EPROM, floppy disc memory, hard disc memory, bubble memory and other types of memory. (20, 21)

COMPUTER SOFTWARE

18. Define and distinguish between different categories of computer software (e.g., system software and application software). (22, 4)
19. Demonstrate an understanding of the general capabilities and limitations of computer software. (22, 23)

HISTORICAL DEVELOPMENT

20. Trace the historical development of computers and be able to discuss the resulting societal effects of the past, present and future. (24, 25)
21. Identify several early computing devices and understand the significance on each to modern computers. (24, 25)

22. Identify and distinguish between the characteristics of the 4 major stages of computer development: vacuum tubes, transistors, integrated circuits, microprocessors. (26)

THE OPERATION OF COMPUTERS

23. Generalize common characteristics of operating a variety of computer systems. (27)
24. Demonstrate the ability to operate a microcomputer system (e.g., on-off sequences, loading, running, saving, copying, and printing). (28)
25. Demonstrate the appropriate use of system commands such as DOS commands: CATALOG, LOAD, RUN, LIST, SAVE, RENAME, DELETE, LOCK, UNLOCK. (29)

SECTION IV: COMPUTER APPLICATIONS

Computer applications refer to the ability to responsibly evaluate, select, and implement a variety of practical computer applications to do meaningful and efficient work based on an understanding of the following: general types of applications, capabilities and limitations of applications, societal impact (past, present, and future), evaluation and selection techniques, and specific applications (word processing, data base management, spreadsheet/financial management, statistical analysis, graphics, and educational applications).

GENERAL APPLICATIONS

1. Identify general types of computer applications and give examples of how the applications can be used in various disciplines or professions (e.g., business, industry, medicine, education, recreation, government): (30)
 - a. information storage and retrieval-record keeping, data base management
 - 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
 - j. computer-assisted instruction
 - k. computer-managed instruction
 - l. problem solving
 - m. forecasting
 - n. market research
 - o. game playing
2. Demonstrate an understanding of the idea that computer applications are generally useful for effective information-processing tasks that require any of the following: (31)
 - a. handling large amounts of information (searching, sorting, deleting, updating, summarizing)
 - b. rapid handling of information
 - c. accuracy
 - d. repetition

- e. the storage and retrieval of information in an accessible form
- 3. Identify some of the factors to consider when making decisions regarding the appropriate use of computers for various problem situations: (32)
 - a. cost
 - b. people's attitudes (fears, anxieties)
 - c. availability of suitable hardware and software/applications
 - d. hardware limitations (memory capacity, lack of peripherals)
 - e. complexity of some computer-supported applications
 - f. application appropriateness
 - g. ethical and moral considerations
- 4. Demonstrate an understanding of the idea that an operator controls the computer system rather than the system controlling the operator (e.g, computers do not make mistakes, people make the mistakes). (33)

SOCIETAL IMPACT

- 5. Demonstrate an understanding of past, present, and future applications and implications of computers in relation to the impact on society. (no specific item)
- 6. Demonstrate an understanding of legal and ethical computer related issues (e.g., copyright law, computer crime, privacy, security) and make responsible decisions based on this understanding. (34)
- 7. Identify computer-related occupations and jobs and demonstrate an awareness of the job description and training for each. (35, 36)

SPECIFIC APPLICATIONS

- 8. Define word processing and understand the common capabilities and limitations as well as the advantages and disadvantages of using word processing. (37)
- 9. Demonstrate an understanding of the utilization of word processing software programs to create, edit, save, and print multi-page documents regularly for academic or personal uses. (38)

10. Define data base management and understand the common capabilities and limitations as well as the advantages and disadvantages of using data base management software. (39, 40)
11. Demonstrate an understanding of the utilization of a data base management system to create, search, sort, retrieve, edit, and print files. (39, 40)
12. Define spreadsheet/financial management software and understand the common capabilities and limitations as well as the advantages and disadvantages of using spreadsheet/financial management software. (41, 42)
13. Demonstrate an understanding of the utilization of a spreadsheet software system. (41, 42)
14. Define and demonstrate an understanding of the utilization of simple statistical programs. (43, 44)
15. Demonstrate an understanding of the uses of computer-generated graphics software. (43)

EDUCATIONAL APPLICATIONS

16. Identify the major uses of computers in education (computer-managed instruction, computer-assisted instruction, educational games, word processing, teacher utility, data management, administrative uses) (45, 46)
17. Identify and demonstrate an understanding of the characteristics of categories of computer-assisted instruction software. (46)
18. Demonstrate an understanding of the functioning of CMI systems. (45, 47)
19. Demonstrate an understanding of the uses of simulations as teaching tools. (46)
20. Identify computer languages well suited to educational applications (e.g., authoring languages for writing interactive instructional programs). (48)
21. Demonstrate an understanding of the capabilities of using LOGO in educational situations. (49)
22. Demonstrate the ability to evaluate the effectiveness of instructional computer programs. (50, 52)
23. Demonstrate the ability to design and/or select and appropriately use computer programs to enrich and extend the regular course instruction. (51, 52)

EVALUATION AND SELECTION

24. Determine what computer hardware/software features are necessary to solve particular problems or suit certain applications. (53, 54, 55, 56)
25. Identify computer hardware/software evaluation criteria, and demonstrate the ability to evaluate and select appropriate computer hardware and software in terms of usefulness to particular applications or problems. (52, 53, 54, 55, 56, 57)

SECTION V: COMPUTER PROGRAMMING

Computer programming refers to the ability to direct the operation of the computer through the skilled use of programming languages. This requires an understanding of problem solving strategies, algorithms and flowcharts, languages, and programming skills.

PROBLEM SOLVING STRATEGIES

1. Demonstrate an understanding that programming is a problem-solving process. (58)
2. Demonstrate the ability to analyze problems and select and/or develop appropriate problem solving strategies for their solution. (72)

ALGORITHMS/FLOWCHARTS

3. Define and demonstrate an understanding of algorithms and flowcharts. (59, 60)
4. Read an algorithm or flowchart and identify the output. (61, 65)
5. Identify and demonstrate an understanding of the function of the symbols used to develop a flowchart. (62, 63)
6. Demonstrate an understanding of why flowcharts are helpful in developing computer programs. (64)
7. Demonstrate the ability to develop algorithms and flowcharts for the solutions to problems. (65, 72)

LANGUAGES

8. Define and distinguish between the different types of languages: machine, assembly, high level, compiler, interpreter). (66, 67)
9. Identify high level programming languages and distinguish between their characteristics in relation to their appropriateness for various applications and grade levels. (e.g., BASIC, PASCAL, COBOL, FORTRAN, LOGO) (68, 69)
10. Demonstrate an understanding of the use of authoring languages to write interactive instructional programs. (70)

PROGRAMMING

11. Define and be able to use computer programming terminology (e.g., variable, string variable, linear programming, branched programming, bug, debugging, coding, statement, syntax, looping, unconditional branch, conditional branch, pseudocode, recursion). (71)
12. Identify the steps in creating a computer program, from problem definition through to implementation. (58, 72)
13. Identify and be able to interpret user-oriented documentation for computer programs. (76)
14. Identify the characteristics of structured/modular programming style. (72, 74)
15. Read a simple program, demonstrate an understanding of the statements and logic of the program, and identify the output. (73, 74, 75)
16. Identify characteristics of well-written computer programs. (74, 76)
17. Demonstrate the ability to translate a given algorithm or flowchart into a high-level programming language. (77)
18. Given a program (or flowchart) containing errors (syntax and logic), identify the error and debug the program so it will execute properly. (78)
19. Modify computer programs written by others. (79)
20. Demonstrate the ability to write simple programs in a high-level language. (e.g., BASIC, PASCAL, LOGO) (78, 79, 80)

APPENDIX F:
COMPUTER LITERACY ASSESSMENT INSTRUMENT

SECTION I: BACKGROUND INFORMATION

Instructions: The first sixteen items concern background information about yourself. The responses to these items will not effect your overall test score. Please fill in the appropriate circle on the answer sheet labeled Sections I and II. Select only one answer for each question.

1. Are you male or female?
 - a. male
 - b. female

2. What is your academic status?
 - a. high school student
 - b. college freshman
 - c. college sophomore
 - d. college junior
 - e. college senior
 - f. graduate student
 - g. not in school
 - h. other

3. What is your area of study in school or work?
 - a. education
 - b. engineering
 - c. science/math
 - d. humanities/liberal arts
 - e. agriculture
 - f. business
 - g. home economics
 - h. computer science
 - i. child development
 - j. other

4. What is your age?
 - a. 14-18
 - b. 19-23
 - c. 24-28
 - d. 29-33
 - e. 33-37
 - f. 38-42
 - g. 43-47
 - h. 48-52
 - i. 53-57
 - j. 58 or older

5. Have you ever taken a non-programming computer course?
 - a. yes
 - b. no

6. If yes, where?
 - a. high school
 - b. computer store
 - c. community college
 - d. university
 - e. hobbyist group
 - f. professional organization
 - g. teacher inservice
 - h. two or more of the above
 - i. other

7. Have you ever taken a computer programming course?
- a. yes
 - b. no
8. If yes, where?
- a. high school
 - b. computer store
 - c. community college
 - d. university/college
 - e. hobbyist group
 - f. professional organization
 - g. teacher inservice
 - h. two or more of the above
 - i. other
9. Have you ever written a computer program?
- a. Yes
 - b. no
10. If yes, what language or languages did you use? (select only one)
- a. BASIC
 - b. COBOL
 - c. FORTRAN
 - d. PASCAL
 - e. LOGO
 - f. PILOT
 - g. PL-1
 - h. ADA
 - i. two or more of the above
 - j. other
11. If you have written computer programs before, how would you rate your programming ability compared to other people you know who have written computer programs?
- a. well below average
 - b. below average
 - c. average
 - d. above average
 - e. well above average

How would you rate yourself compared to the average persons you encounter in relation to the following areas?

12. Overall skills in the use of computers.
- a. well below average
 - b. below average
 - c. average
 - d. above average
 - e. well above average

13. Confidence in your own ability¹⁹⁷ to use computers.
- a. well below average
 - b. below average
 - c. average
 - d. above average
 - e. well above average
14. Natural talent for the use of computers.
- a. well below average
 - b. below average
 - c. average
 - d. above average
 - e. well above average
15. Willingness to use a computer.
- a. well below average
 - b. below average
 - c. average
 - d. above average
 - e. well above average
16. How well you like to use a computer.
- a. well below average
 - b. below average
 - c. average
 - d. above average
 - e. well above average

198
SECTION II: COMPUTER ATTITUDES

Instructions: Please indicate how you feel about the following statements. Use the scale below to indicate your feelings. Mark the appropriate circle on the answer sheet labeled Sections I and II.

- | | |
|--------------------|-----------------------|
| 1 = Strongly agree | 4 = Slightly disagree |
| 2 = Agree | 5 = Disagree |
| 3 = Slightly agree | 6 = Strongly Disagree |

Part One (Ellsworth and Bowman, 1982)

- | | |
|--|-------------|
| 17. A person today cannot escape the influence of computers. | 1 2 3 4 5 6 |
| 18. Computers are beyond the understanding of the typical person. | 1 2 3 4 5 6 |
| 19. Credit rating data banks are a worthwhile use of computers. | 1 2 3 4 5 6 |
| 20. Our country would be better off if there were no computers. | 1 2 3 4 5 6 |
| 21. Computers make mistakes at least 10% of the time. | 1 2 3 4 5 6 |
| 22. Computers are a tool, just like a hammer or lathe. | 1 2 3 4 5 6 |
| 23. Computers will improve health care. | 1 2 3 4 5 6 |
| 24. Someday I will have a computer, or a computer terminal in my home. | 1 2 3 4 5 6 |
| 25. Programmers and operators make mistakes, but computers are, for the most part, error free. | 1 2 3 4 5 6 |
| 26. Computers slow down and complicate simple business operations. | 1 2 3 4 5 6 |
| 27. Computers will improve law enforcement. | 1 2 3 4 5 6 |
| 28. A computer may someday take my job. | 1 2 3 4 5 6 |
| 29. Computers isolate people by preventing normal social interactions among users. | 1 2 3 4 5 6 |
| 30. It is possible to design computer systems which protect the privacy of data. | 1 2 3 4 5 6 |
| 31. Computers will replace low-skill jobs and create jobs needing specialized training. | 1 2 3 4 5 6 |

1 = Strongly agree
 2 = Agree
 3 = Slightly agree

4 = Slightly disagree
 5 = Disagree
 6 = Strongly Disagree

32. Computers will improve education. 1 2 3 4 5 6
33. Computers will create as many jobs as they eliminate. 1 2 3 4 5 6

Part Two (Maurer, 1983)

34. Having a computer available to me would improve my productivity. 1 2 3 4 5 6
35. If I had to use a computer for some reason, it would probably save me some time and work. 1 2 3 4 5 6
36. If I use a computer, I could get a better picture of the facts and figures. 1 2 3 4 5 6
37. Having a computer available would improve my general satisfaction. 1 2 3 4 5 6
38. Having to use a computer could make my life less enjoyable. 1 2 3 4 5 6
39. Having a computer available to me could make things easier for me. 1 2 3 4 5 6
40. I feel very negative about computers in general. 1 2 3 4 5 6
41. Having a computer available to me could make things more fun for me. 1 2 3 4 5 6
42. If I had a computer at my disposal, I would try to get rid of it. 1 2 3 4 5 6
43. I look forward to a time when computers are more widely used. 1 2 3 4 5 6
44. I doubt if I would ever use computers very much. 1 2 3 4 5 6
45. I avoid using computers whenever I can. 1 2 3 4 5 6
46. I enjoy using computers. 1 2 3 4 5 6
47. I feel that there are too many computers around now. 1 2 3 4 5 6
48. Computers are probably going to be an important part of my life. 1 2 3 4 5 6

1 = Strongly agree
 2 = Agree
 3 = Slightly agree

4 = Slightly disagree
 5 = Disagree
 6 = Strongly Disagree

- | | |
|--|-------------|
| 49. A computer could make learning fun. | 1 2 3 4 5 6 |
| 50. If I were to use a computer, I could get a lot of satisfaction from it. | 1 2 3 4 5 6 |
| 51. If I had to use a computer, it would probably be more trouble than it was worth. | 1 2 3 4 5 6 |
| 52. I am usually uncomfortable when I have to use computers. | 1 2 3 4 5 6 |
| 53. I sometimes get nervous just thinking about computers. | 1 2 3 4 5 6 |
| 54. I will probably never learn to use a computer. | 1 2 3 4 5 6 |
| 55. Computers are too complicated to be of much use to me. | 1 2 3 4 5 6 |
| 56. If I had to use a computer all the time, I would probably be very unhappy. | 1 2 3 4 5 6 |
| 57. I sometimes feel intimidated when I have to use a computer. | 1 2 3 4 5 6 |
| 58. I sometimes feel that computers are smarter than I am. | 1 2 3 4 5 6 |
| 59. I can think of many ways that I could use a computer. | 1 2 3 4 5 6 |

SECTION III: COMPUTER SYSTEMS

Instructions: Read each question carefully and then select the most appropriate answer from the five choices and mark the appropriate circle on the answer sheet labeled Sections III, IV, and V. If you do not know the answer, try to make an educated guess if possible, otherwise leave the item blank.

1. Which of the following is the primary reason that program instructions and data are handled by modern digital computers in binary form?
 - a. A given value may be represented in binary form using fewer place values than would be required in base ten.
 - b. Binary numbers are easier for the operator to enter into the keyboard than are base ten numbers.
 - c. It is simplest to design circuits which operate in only two logical states rather than ten separate states.
 - d. Binary numbers more accurately represent logical operations than would the far preferable base 8 system.
 - e. The binary number system is a traditional, though unnecessary, holdover from the days of vacuum tube technology.

2. Computer systems are commonly used to perform "data processing" functions. This term may best be described as
 - a. the process of critically analyzing large sets of data and making subjective decisions based on that data.
 - b. a type of information management used primarily in business and government applications, usually involving statistical operations.
 - c. the exclusive domain of mainframe computers--data processing is beyond the capabilities of a microcomputer because of its limited memory.
 - d. the process of handling information, including such operations as sorting, calculating, recording, classifying, and summarizing.
 - e. the process of adding, subtracting, multiplying, and dividing numbers in base two.

3. Place in order from first to last the operations that take place as a problem is being solved with the aid of a computer.
 1. print a report
 2. read data into the computer
 3. develop and program an algorithm
 4. calculate results
 5. code the data onto input medium
 - a. 1,2,3,4,5
 - b. 2,1,5,4,3
 - c. 3,5,2,4,1
 - d. 5,4,3,2,1
 - e. 3,5,4,2,1

4. The major purpose of a computer software program is to
 - a. supply instructions to the computer.
 - b. read punched cards into the computer.
 - c. develop an algorithm for problem solving.
 - d. design input data for the computer.
 - e. output the results of the operation of the computer.

5. Computer hardware represents only a portion of the cost of a complete computer system because
 - a. disk drives, printers and other peripheral devices are quite expensive.
 - b. quality computer systems, such as the Apple IIe or the IBM PC, require extra interface cards and controller cards in order to be fully functional.
 - c. a computer's true cost must be weighed against the eventual savings in time and human resources which the computer makes possible.
 - d. the operating system and other machine language programs resident in ROM must be obtained at extra cost.
 - e. computer hardware cannot function without adequate software, which represents an additional expense.

6. Computers and certain computer peripherals may be classified as either digital or analog devices. Which of the following groups includes exclusively digital hardware devices?
 - a. CPU, RAM chip, game paddle
 - b. ROM chip, serial interface card, microprocessor
 - c. CPU, compiler, word processing program
 - d. RAM chip, ROM chip, operating system
 - e. BASIC, integrated circuit, interpreter

7. Batch processing refers to a processing mode in which
 - a. a program is run with direct interaction between the computer and the user, usually with the program on magnetic tape or disk.
 - b. a batch of data is collected over an extended period and then processed concurrently using multiple processing units.
 - c. a program is run without interaction between the computer and the user. The program with its data is submitted to the computer usually on punched cards, and results are then inserted as required by the program.
 - d. many computers are networked together so many programs can be processed at one time.
 - e. microcomputers are connected to mainframe computers in a time-sharing situation.

8. Which of the following groups of computer terms does not refer exclusively to computer hardware?
- CRT, CPU, RAM chip
 - speech synthesizer, disk drive, graphics digitizing pad
 - letter quality printer, ROM chip, I/O connector
 - integrated circuit, BASIC, diskette, power supply
 - keyboard, disk drive, video monitor
9. Which of the following best depicts a special purpose computer system?
- microcomputers in a network configuration for classroom use
 - a mainframe computer with time-share terminals
 - a personal computer with a printer for wordprocessing
 - a climate control computer for a building
 - a minicomputer with dedicated terminals
10. Which of the following groups of computer hardware and software are representative characteristics of a MICROCOMPUTER system?
- 5 1/4 inch floppy disk, 5 megabytes of read only memory, card reader
 - microprocessor, BASIC, 5 1/4 inch floppy disk
 - dual disk drives, microprocessor, time-sharing system
 - BASIC, 5 megabytes of read only memory, time sharing system
 - card reader, microprocessor, 5 megabytes of read only memory
11. Which of the following best reflects the relationship between microcomputers, minicomputers, and mainframe computers in terms of the average memory capacity? (ordered from least to greatest average memory capacity)
- mainframe, microcomputer, and minicomputer
 - microcomputer, mainframe, minicomputer
 - microcomputer, minicomputer, mainframe
 - They all are capable of having equal memory capacity.
 - The memory capacity depends on the cost of the system, so it is not possible to order them by category of computer.
12. Several microcomputers connected together with communication lines in order to access the same programs and data is an example of
- time-sharing.
 - multipleprocessors.
 - networking.
 - interface interaction.
 - modulation-demodulation.

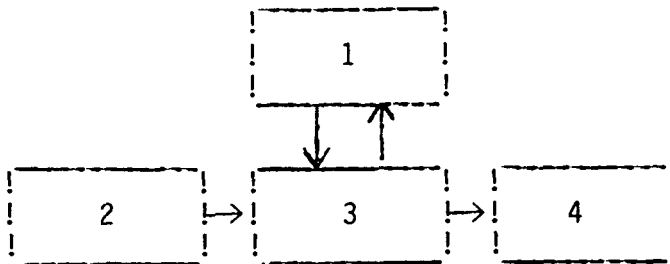
13. Serial communication refers to
 - a. transferring information from one computing device to another eight bits at a time.
 - b. transferring information from one computing device to another one bit at a time.
 - c. communicating with the computer via a series of bit to byte interactions.
 - d. communicating with the computer via a series of programming statements.
 - e. transferring information from the program to the central processor one bit at a time.

14. Which of the following is a common function of operating systems?
 - a. Providing an orderly and consistent input/output environment for the various elements of the computer.
 - b. Permitting compatibility among all microcomputers, regardless of the microprocessors they incorporate.
 - c. Controlling the voltage levels supplied by the power supply.
 - d. Determining the number of bits in each of the computer's data words.
 - e. Providing a list of user-friendly commands so the user can operate the system.

15. Computer hardware failures are most often caused by
 - a. dusty operating environment.
 - b. cold operating environment.
 - c. physical abuse of the hardware.
 - d. power line spikes, dropouts, and surges.
 - e. defective software.

16. Which of the following is not necessary for the proper care and maintenance of computer systems?
 - a. maintaining a dust-free operating environment
 - b. maintaining a room temperature of at least 68 degrees
 - c. the use of voltage-controlled and filtered circuits when supplying power to the computer.
 - d. maintaining a relative humidity of 40 - 60% to minimize static electricity
 - e. providing adequate air circulation around the computer

17. A microcomputer system is not well suited for performing complex statistical functions on large data sets because
- the built-in video displays of most microcomputers would be too small to show the many values and formulas in a typical applications software package.
 - microcomputer systems generally do not have sufficient memory for storage of elaborate programs and large amounts of data.
 - no microcomputer system can handle such functions. To do any kind of number-crunching work, you need a larger computer.
 - microcomputers are not equipped with disk drives and thus cannot load large statistical programs from disks.
 - microcomputers represent an inexpensive, very limited class of computers and are good for little more than arcade-type games.
18. A primary function of the Central Processing Unit of a computer is to _____ information.
- store
 - input
 - output
 - input/output
 - analyze/manipulate
19. Below is a block diagram of a computer system with the arrows indicating the sequence of data movement within the system. Which of the following are correct labels for the components?



- | | |
|--|---|
| <ol style="list-style-type: none"> 1-memory
2-input
3-processor unit
4-output 1-video display
2-keyboard
3-disk drive
4-printer 1-arithmetic/logic unit
2-control unit
3-operating system
4-input/output device | <ol style="list-style-type: none"> 1-input/output device
2-CPU
3-RAM
4-ROM 1-processor unit
2-input device
3-memory
4-output device |
|--|---|

20. Which of the following statements is not true concerning RAM and ROM?
- Information stored in RAM can be changed by the user, while information stored in ROM cannot be changed by the user.
 - Information stored in both RAM and ROM will be destroyed if the power to the computer is turned off.
 - ROM stores the control program of the computer.
 - The amount of RAM a computer determines the memory density of the computer.
 - RAM is volatile and ROM is nonvolatile.
21. The type of memory that is most likely to be of interest to a prospective microcomputer buyer (for the reason given) is:
- EPROM, because the ability to erase and reprogram memory is needed if one is to run application programs.
 - PROM, because all memory units in a computer are erased each time the power is shut off or a new disk is booted.
 - ROM, because the amount of ROM in a computer determines the size of programs a computer can run and the amount of data which can be stored on disks.
 - RAM, because a computer with an insufficient amount of RAM may not be able to load and run some of the application programs.
 - Stringy-floppy storage, because the ability to link (string) files is important to many microcomputer users.
22. A disk operating system is a special category of software that allows the computer to
- use magnetic disks for long term memory storage.
 - operate disks that allow several computers to be connected to one another.
 - receive disk information from devices such as modems.
 - present computer disk operation information to the user.
 - expand the usefulness of Read Only Memory in a disk.
23. Which of the following statements concerning computer software is false?
- Computer software could be defined as the programs, procedures and associated documentation concerned with the operation of computer hardware systems.
 - A program written in BASIC to add numbers together and print their total is an example of computer software.
 - The information coded on Read Only Memory chips in a microcomputer is actually software.
 - Lists of instructions to the computer are called software.
 - When peripheral boards and other devices intended to expand a computer's capabilities are added to a microcomputer they are defined as software expansions.

Questions 24 and 25 each describe a historical computing device. Identify the correct device for each description.

24. This item was the first device that used punched cards of instructions to control the operation on a machine.
 - a. Mark I
 - b. Analytical engine
 - c. ENIAC
 - d. Hollerith's tabulating machine
 - e. Jacquard's loom

25. This device was the first automatic electronic digital computing device to be developed, but did not receive recognition until years later.
 - a. UNIVAC
 - b. Atanosoff-Berry Computer (ABC)
 - c. EDVAC
 - d. Analytical engine
 - e. Pascal's calculating machine

26. Four generations of modern computers can best be characterized by which group of words?
 - a. punched cards, printed lines, control panels, diskettes
 - b. math tables, difference engine, analytical engine, calculator
 - c. vacuum tubes, transistors, integrated circuits, microprocessors
 - d. Aiken, Mauchly, Eckert, Jobs
 - e. relays, electromechanical, vacuum tubes, transistors

27. Formatting a magnetic floppy diskette is the process of
 - a. telling the computer how to set the top and side margins for final printing of a document.
 - b. copying a set of programs you have written onto a backup data disk.
 - c. checking to see if the disk you have purchased is the proper size for your computer's disk drive.
 - d. organizing the disk into tracks and sectors to enable te computer to store information on it.
 - e. instructing the disk drive to accept the diskette.

28. You have inserted a disk into the disk drive of a microcomputer. What is the usual next step in running a program stored on the disk?
 - a. type the command that results in a listing of the program statements.
 - b. type the command that results in the program being loaded into the computer's memory.
 - c. type the command the results in the execution of the program.
 - d. type the command that results in saving the program on the disk.
 - e. type the command that "boots" the disk operating system.

29. Which of the following DOS commands would you expect to result in a list of programs on a disk?
- a. LIST
 - b. RUN then LIST
 - c. UNLOCK, then RUN, then LIST
 - d. CATALOG
 - e. LOAD CATALOG, then RUN

SECTION IV: COMPUTER APPLICATIONS

30. Artificial intelligence is being used in which of the following ways?
- the industrial field in the use of robots on assembly lines
 - the business field to collect and sort data
 - the education field to teach individualized lessons
 - the music industry to produce and record sounds
 - the medical field to diagnose illness and prescribe treatments
31. Which of the following is the least accurate characteristic of a task appropriate for a computer application?
- requires rapid processing of information
 - involves repetitious operations
 - involves manipulating large amounts of information
 - requires continuous interaction with the user
 - requires ease in storing and retrieving data
32. Which of the following factors would you consider least important to consider when making a decision whether to use a computer to perform a specific task?
- Does the computer system available possess the appropriate hardware requirements to perform the necessary operations?
 - Is the computer the most appropriate tool to perform the task?
 - Is an appropriate program available that is compatible with the computer hardware system that is available?
 - What knowledge and skills are required of the computer user to implement the application?
 -
33. Consider the following situation:
- Mr. Brown received a computer generated bill for \$37.50 for merchandise he bought on credit. However, he has already paid the full amount.
- What is the most likely cause of the error?
- computer hardware malfunction
 - human error
 - printer malfunction
 - tape or disk damage
 - power surges in the computer

34. Which of the following does not describe a case of computer crime?
- Bill purchased a copyrighted computer program for figuring his income tax. His friend, John expressed a desire to use the program, so Bill copied it for him and gave it to him for Christmas.
 - Sam used a password to gain access to the computer owned by his former employer. Over long distance phone lines, he used the computer to direct the operation of machines owned by his present employer.
 - An employee of a Motor Vehicle Department added 1000 names to a computerized list of approved applicants for driver's licenses. The employee then sold the licenses.
 - A computer analyst at a Wall Street brokerage house programmed a computer to sell nonexistent securities through fictitious accounts.
 - A school teacher who used a computer in the classroom wrote her own educational programs and then made copies of them to sell to other teachers.

35. Which of the following computer-related job titles is the most appropriate for this job description?

Starts the computer system when necessary.
Mounts tapes or disks to provide computer with proper data.
Loads programs into the computer.
Performs sequential activities necessary to run programs.
Supplies printers with paper and ribbons.

- Applications programmer
- Systems Analyst
- Systems programmer
- Computer scientist
- computer operator

36. Which of the following computer-related job titles is the most appropriate for this job description?

Knows one or more computer languages.
Writes flowcharts and instructions for programs for computer users.
Tests and revises programs until correct.
Explains programs to users through documentation.

- applications programmer
- system analyst
- system programmer
- computer scientist
- computer operator

37. One of the main advantages of writing with any word processing system over writing with a typewriter, is that during editing and correction
- print quality is improved.
 - hyphenation of words is automatic.
 - the unchanged portions of the text need not be retyped by hand.
 - different display systems can be used.
 - spelling can be corrected automatically.
38. Fred has a letter on file in which he wants to change the name, address, and date before making a printout. Using a word processing system, what would be the most likely sequence of his actions?
- Load, edit, save, print
 - Load, save, print, edit
 - Edit, save, print, reedit
 - Edit, save, print, run
 - load, print, save, edit, reprint
39. Which of the following is not a function of most data base management programs?
- The ability to create a new file
 - The ability to add, delete, or change records within a file
 - The ability to word process documents
 - The ability to sort file records
 - The ability to retrieve records from a file
40. Which of the following is the most accurate description of a use for any data-base management computer software program?
- With a data-base management system, data can be analyzed statistically.
 - With a data-base management system information is organized and stored efficiently so retrieval is faster and more reliable than manual filing systems.
 - Term papers can be written using data base management programs and stored for later retrieval and printing.
 - Interactive educational lessons can be written and stored using data base management systems.
 - Programs that teach problem solving skills can be written with a data base management system.
41. Mike wants to purchase a software package for his personal computer that can be used to create templates for projecting production costs on his dairy farm. What type of software package would be the most appropriate for his needs?
- spreadsheet
 - data base management
 - word processing
 - statistical
 - graphics

42. Spreadsheet programs can be used to create
- business letters with extra wide horizontal margins.
 - files that can keep track of and sort information about students such as student attitudes and attendance.
 - a personal budget that can be used for projections of savings and expenditures.
 - a mailing list that can be sorted alphabetically.
 - extra wide lesson plans.
43. Which of the following types of computer applications would be the most appropriate for visualizing the percentage of people voting for each candidate in an election
- word processor
 - spreadsheet
 - data base management
 - computer graphics
 - statistical package
44. Which of the following types of computer applications would be the most appropriate for analyzing the amount of difference between students' scores on a test?
- word processor
 - spreadsheet
 - data base management-grade book
 - artificial intelligence
 - statistical package
45. Ms. Jones is using a software package that will list class scores and average scores. It can also be used to assist the instructor in choosing various lessons for different achievement levels. This is an example of using the computer for
- Computer-managed instruction.
 - Computer-assisted instructon.
 - tutorial programs.
 - simulation programs.
 - strengthening problem solving skills.
46. Which type of computer software is the most capable of recreating situations such as the electoral process, the operation of nuclear power plants, and the lunar landing to allow students to experience the situation?
- administrative
 - tutorial
 - simulation
 - computer managed instruction
 - drill and practice

47. Which of the following best describes an application of computer managed instruction?
- Mr. Smith uses a computer in his classroom as a reward for students who complete their work quickly and accurately.
 - Ms. Jones uses a computer and a software program that keeps track of course goals and objectives, student grades, student progress through lessons, and prescribes the instructional program for individual students.
 - Ms. Johnson uses a computer in a business class to provide drill and practice for touch typing.
 - Mr. Brown uses a computer and a simulation program to teach his science class about the space program.
 - Principal Anderson uses an electronic spreadsheet to keep track of the school budget.
48. As an elementary teacher, with little programming experience, you have a gifted student needing accelerated lessons. Which of the following languages would be the most appropriate for you to use to write a simulation?
- FORTRAN
 - COBOL
 - BASIC
 - PILOT
 - PASCAL
49. When programming with LOGO, a student is not capable of
- directing a computer-generated turtle to draw on a video screen.
 - writing BASIC programs that display graphics.
 - programming music to accompany graphic displays.
 - writing stories and printing them on the screen or paper.
 - programming the turtle to do recursive actions.
50. As a teacher, which of the following programs would you expect to be the most effective computer-assisted instruction program to teach a new concept to students with a wide range of abilities.
- the program presents questions about the concept repeatedly until the student answers correctly.
 - the program presents large amounts of information for the student to read, and then quizzes them over their comprehension of the information.
 - the program presents information in a linear programming fashion, so all students receive the same information in the same order.
 - the program presents information in a branching programming fashion, so students receive information based on their individual responses.
 - the program contains many color graphics and makes sounds for reinforcement of correct answers.

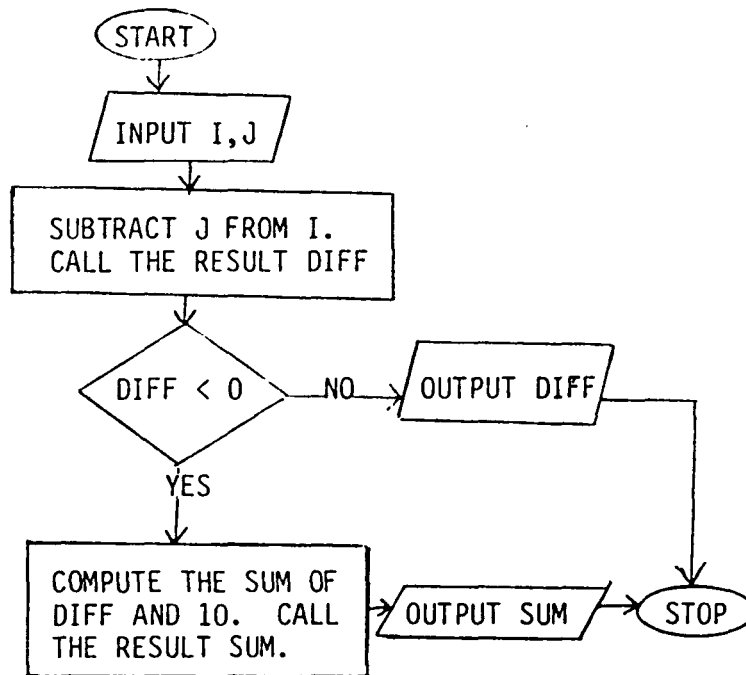
51. Which of the following would be the most appropriate first step in the development of a computer-assisted instruction program for use in a classroom?
- flowchart the lesson
 - identify and organize the content to be taught
 - select an instructional strategy
 - identify specific objectives
 - program the lesson with a programming language
52. Which of the following is the least important criterion to consider when evaluating and selecting a computer-assisted instruction program to teach a particular concept to a particular student or group of students?
- Does the program utilize the unique capabilities of the computer?
 - Is the content accurate and properly sequenced?
 - Does the program provide positive reinforcement and feedback to the students?
 - Does the program keep record of the students' correct responses?
 - Is the language used appropriate for the abilities of the students?
53. Mary wants to write some letters to prospective employers. She would also like to keep files on each of these employers detailing the size of the company, the type of position, and the job offers. Which personal computer hardware setup contains only the essential components to meet her needs?
- computer keyboard, printer, monitor, disk drive
 - computer keyboard, disk drive
 - computer keyboard, disk drive, printer
 - printer, computer keyboard
 - monitor, disk drive, printer
54. What software is the most appropriate to meet Mary's needs?
- a word processing package
 - a data base management package
 - both a word processing and a data base management package
 - both a word processing and a spreadsheet package
 - both a spreadsheet and a data base management package

55. As a special education coordinator for math, every three weeks you receive a list of students from all the math teachers, grades 3-6, involved with remedial math sections. These lists state which modules their students have passed. You want to find a computer software package to print and update a master record record of all students on the individual lists. Which feature would be least essential to have in the software package you'd select?
- An arithmetic option that allows computing the average for a set of numeric data.
 - An option that allows files/records to be changed nonsequentially.
 - An option that allows files/records to be sorted numerically.
 - An option for sorting files/records character by character.
 - The ability to select and print a series of individual files/records.
56. As the owner of a small business, you have decided to use a microcomputer for word processing of such items as invoices, letters to customers, billing information, and annual reports. Which of the following hardware devices would not be essential to accomplish your needs?
- a typewriter keyboard
 - a disk or tape drive
 - a video display
 - a dot matrix printer
 - a letter quality printer
57. As the director for computing services for a large business, what of the following criteria would you consider to be the least important when evaluating and selecting computer software programs for a specific purpose, such as record keeping, that will be used by a large number of personnel?
- Is the program written in a user-friendly format?
 - If it possible to easily modify the program to better meet the needs of the users.
 - How much time and energy will be required for the personnel to learn to use the software accurately.
 - What are the needs of the computer users, and does the program meet those needs?
 - Is the program written in a computer language that most of the users are proficient at?

SECTION V: COMPUTER PROGRAMMING

58. Both a computer programmer and a computer user must understand what the problem to be solved by the computer is and how it is to be solved. In what step in the problem-solving process is this usually accomplished?
- a. problem definition
 - b. flowcharting
 - c. program writing
 - d. debugging stage
 - e. executing program
59. A graphic interpretation of a solution process that sequentially depicts, in words and symbols, each step needed to solve a specific problem is
- a. an algorithm.
 - b. a computer program.
 - c. a flowchart.
 - d. a problem statement.
 - e. a BASIC language program.
60. An algorithm is
- a. a graphic representation of a solution process.
 - b. an instrument of communication used by all those concerned with computer processing.
 - c. a programming language that requires adherence to syntax and structure rules.
 - d. a set of rules designed to solve in a finite number of steps.
 - e. a set of instructions written in a language that the computer can understand.

Question 61 concerns the following diagram.



61. If the values of I and J are 8 and 5 respectively, what value will be output at the end of execution?
- 3
 - 3
 - 7
 - 13
 - 13

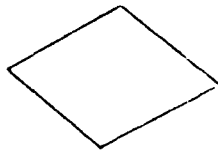
For questions 62-63 match the flowcharting symbol with the appropriate function.

62.



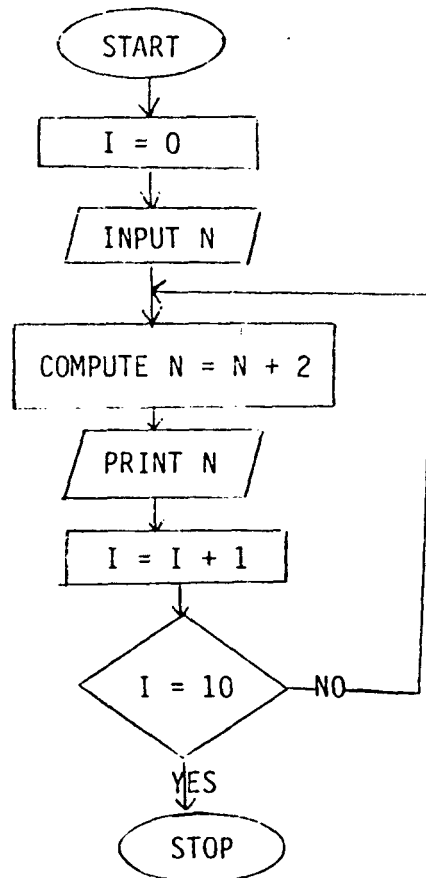
- tests a condition
- input/output of information
- processes information
- starts or ends the program
- branches to another part of the program

63.



- a. loops to a previous section of the program
 - b. tests a condition
 - c. processes information
 - d. starts or ends the program
 - e. input/output of information
64. For the computer programmer, which of the following is not a direct benefit derived from flowcharting the solution to an algorithm before coding a computer program?
- a. The elimination of logic errors.
 - b. Avoiding replications of blocks of code.
 - c. Getting an indication of what elements must be included in a comprehensive test data file.
 - d. Facilitating the integration of computer tasks into an executable computer program.
 - e. The elimination of syntax errors in the coding of the program.

Item 65 refers to the following diagram.



65. Assume N is positive. Which algorithm below is the origin of the diagram above?
- Input an integer N and output the sum of the next ten consecutive integers.
 - Input an integer N and output the tenth consecutive even integer greater than N .
 - Input an integer N and output the next ten consecutive even integers if N is even, or the next ten consecutive odd integers if N is odd.
 - Input an integer N and output a series of integers until the value of N is 10.
 - Input an integer N and output N and the next ten consecutive even integers greater than N .

For questions 66 and 67, match the examples of computer program segments with the type of programming language it is written in.

66. 10 T = 0
 20 K = 1
 30 T = T + 1
 40 K = K * T
 50 IF 1 < 100 THEN 30
 60 PRINT K
 70 END
- a. machine language
 b. assembly language
 c. compiler language
 d. interpreter language
 e. high-level language
67. 7413 13 00 0 000000 0003
 0000 00 00 0 000000
- 0500 00 00 0 000000 0000
 1000 00 00 0 000004 0000
 0100 00 00 0 000001 0000
- a. machine language
 b. assembly language
 c. compiler language
 d. interpreter language
 e. high-level language

For questions 68, 69, and 70 match the description of a programming language with the appropriate language.

68. Structured language, recommended for number crunching, has efficient data structures and is portable from mainframe to microcomputer.
- a. FORTRAN
 b. LOGO
 c. BASIC
 d. PILOT
 e. PASCAL
69. Computer language used often for scientific and engineering applications.
- a. PILOT
 b. LOGO
 c. BASIC
 d. COBOL
 e. FORTRAN

70. Authoring language designed for use by teachers.

- a. PILOT
- b. COBOL
- c. BASIC
- d. FORTRAN
- e. LOGO

Item 71 refers to the following:

```
Read a student record
Do while not end of file
    translate grades to points
    divide to find final average
    print record
End do
```

71. The segment above can best be described as

- a. low level code.
- b. a Pascal program.
- c. a Basic program.
- d. pseudo code.
- e. high level code.

72. The following is a partial problem statement:

Then compare the individual transaction account numbers with those in the company's master file. For any illegal account number (not in the master file) output the account number, name of the client, and the date of transaction.

If a top-down approach to programming were used, what would be the first step taken by the programmer?

- a. Write the code to output the illegal account records.
- b. Write blocks of pseudo-code describing each task that the program will have to execute to achieve the desired output.
- c. Write the code to search the master file and compare account numbers.
- d. Set up the trial data file that will be used to test the program.
- e. Write the code to input the master file.

Items 73 - 80 are based on the following program commands and seven procedures. Study their given descriptions carefully.

:X - a storage variable which may contain characters or numbers

:Y - a storage variable which may contain characters or numbers

SET :X (xxxx) - The command SET stores the contents of the parentheses in the variable :X (or :Y if used instead)

Code to increment a variable counter would be : SET :X (:X + 1)

PRINT :X - The command PRINT outputs the contents of :X (or :Y if used) to the screen.

LOOP 10 - The command LOOP repeats a set of commands the number of times indicated by the number immediately following LOOP (in this case 10 times). The set of commands to be repeated must start with BEGIN and stop with END. For example, code to print "Hello" five times could be:

```
SET :X (Hello)
LOOP 5
  BEGIN
  PRINT :X
  END
```

IF - The command IF allows boolean comparisons (i.e., <, >, =, <>) between variables. If the comparison is true, then the block of code following the IF statement is executed. The block of code must start with a BEGIN and stop with a END.

```
Example
SET :X (5)
SET :Y (1)
LOOP 5
  BEGIN
  IF :Y = 5
  BEGIN
  PRINT :X
  END
  SET :Y (:Y+1)
  END
```

HALT - The use of the command HALT stops execution of the procedure it occurs in.

PROCEDURE name - The command PROCEDURE identifies a block of code which may be executed by calling "name." This block of code must start with a BEGIN and stop with a END. For example, a procedure titled "Fivehello" would be written as :

```

PROCEDURE Fivehello
  BEGIN
  SET :X (Hello)
  LOOP 5
    BEGIN
    PRINT :X
    END
  END
END

```

```

PROCEDURE one
  BEGIN
  SET :X (Hello)
  PRINT :X
  PRINT :X
  PRINT :X
  PRINT :X
  END

```

```

PROCEDURE two
  BEGIN
  SET :X (Hello)
  LOOP 4
    BEGIN
    PRINT :X
    END
  END

```

```

PROCEDURE three
  BEGIN
  SET :X (Y)
  LOOP 10
    BEGIN
    LOOP 10
      BEGIN
      LOOP 10
        BEGIN
        PRINT :X
        END
      END
    END
  END
END

```

```

PROCEDURE four
  BEGIN
  SET :X (0)
  LOOP 10
    BEGIN
    IF :X = 10
      BEGIN
      HALT
      END
    PRINT :X
    SET :X (:X + 1)
    END
  END

```

```

PROCEDURE five
  BEGIN
  PRINT hello
  PRINT how
  PRINT are
  PRINT you
  END

```

```

PROCEDURE six
  BEGIN
  SET :X (hello)
  PRINT :X
  SET :X (how)
  PRINT :X
  SET :X (are)
  PRINT :X
  SET :X (you)
  PRINT :X
  END

```

```

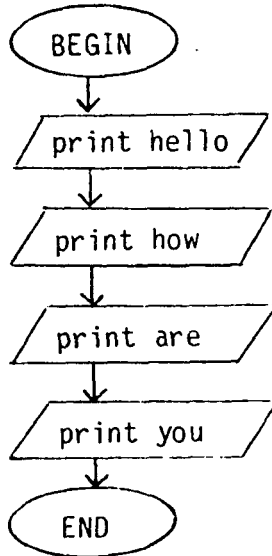
PROCEDURE seven
  BEGIN
  SET :X (bug)
  SET :Y (1)
  PRINT :X
  SET :Y (:Y + 1)
  PROCEDURE seven
  IF :Y > 3
    BEGIN
    HALT
    END
  END
END

```

73. Which of the following is most accurate:
- PROCEDURE one and PROCEDURE two produce the same results.
 - PROCEDURE one and PROCEDURE two produce different results.
 - PROCEDURE one has improper syntax.
 - PROCEDURE two has improper syntax.
 - PROCEDURE one and PROCEDURE two both have improper syntax.
74. In terms of structured programming style, is PROCEDURE one or PROCEDURE two better?
- PROCEDURE one.
 - PROCEDURE two.
 - Because the two procedures produce different results, no comparisons can be made.
 - It depends on the programming style of the programmer.
 - It depends on the type of computer the program is going to be run on.
75. If PROCEDURE three were run, the result would be
- the letter Y being printed 1000 times.
 - the letter X being printed 1000 times.
 - the contents of variable Y being printed 1000 times.
 - the letter Y being printed 100 times.
 - the program would not run because it has a bug.
76. The reason for indenting code, as in PROCEDURE three, is:
- indenting makes the code more readable.
 - indenting is required for the computer to execute structured programming code.
 - indenting discriminates between structured programming and non-structured programming.
 - indenting tells the computer to space the output in a certain way.
 - indenting does not serve a purpose at all.

Question 77 is based on the flowchart below and the PROCEDURES five and six.

FLOWCHART:



77. Which of the following statements is the most accurate?
- PROCEDURE five is the correct code for the flowchart.
 - PROCEDURE six is the correct code for the flowchart.
 - Both five and six would represent the flowchart.
 - Both five and six have syntax errors.
 - Flowcharts are not used in structured programming, so neither procedure is the correct code for the flowchart.

78. PROCEDURE seven was written to print the word "bug" three times. However, there is a bug in the procedure so it doesn't execute correctly. Which of the following procedures is the debugged PROCEDURE seven?

a. PROCEDURE seven
 BEGIN
 SET :X (bug)
 SET :Y (0)
 PRINT :X
 SET :Y (:Y + 1)
 PROCEDURE seven
 IF :Y = 3
 BEGIN
 HALT
 END
 END

b. PROCEDURE seven
 BEGIN
 SET :X (bug)
 SET :Y (1)
 PRINT :X
 SET :Y (:Y + 1)
 PROCEDURE seven
 END

c. PROCEDURE seven
 BEGIN
 SET :X (bug)
 IF :Y = 0
 BEGIN
 SET :Y (1)
 END
 PRINT :X
 SET :Y (:Y + 1)
 IF :Y > 3
 BEGIN
 HALT
 END
 PROCEDURE seven
 END

d. PROCEDURE seven
 BEGIN
 IF :Y = 3
 SET :X (bug)
 SET :Y (0)
 PRINT :X
 SET :Y (:Y + 1)
 PROCEDURE seven
 END

e. PROCEDURE seven
 BEGIN
 IF :Y > 3
 BEGIN
 HALT
 END
 SET :X (bug)
 SET :Y (1)
 PRINT :X
 SET :Y (:Y + 1)
 PROCEDURE seven
 END

79. How would you modify PROCEDURE four so that the even numbers between 0 and 20 would be printed as the output?

- a. PROCEDURE four
 BEGIN
 SET :X (0)
 LOOP 10
 BEGIN
 IF :X = 20
 BEGIN
 HALT
 END
 PRINT :X
 SET :X (:X + 1)
 END
 END
- b. PROCEDURE four
 BEGIN
 SET :X (0)
 LOOP 11
 BEGIN
 PRINT :X
 SET :X (:X + 2)
 END
 END
- c. PROCEDURE four
 BEGIN
 SET :X (20)
 LOOP 20
 BEGIN
 IF :X = 20
 BEGIN
 HALT
 END
 PRINT :X
 SET :X (:X + 1)
 END
 END
- d. PROCEDURE four
 BEGIN
 SET :X (20)
 LOOP 10
 BEGIN
 IF :X = 0
 BEGIN
 HALT
 END
 PRINT :X
 SET :X (:X + 2)
 END
 END
- e. PROCEDURE four
 BEGIN
 SET :X (0)
 PRINT :X
 SET :X (:X + 2)
 PROCEDURE four
 IF :X = 20
 BEGIN
 HALT
 END
 END

80. Write a procedure that uses recursion to produce the same output as PROCEDURE one. Then select the procedure below that is most like the procedure you wrote.

- a. PROCEDURE recursion
 BEGIN
 IF :Y = 4
 BEGIN
 HALT
 END
 SET :X (Hello)
 SET :Y (1)
 PRINT :X
 SET :Y (:Y + 1)
 PROCEDURE recursion
 END
- b. PROCEDURE recursion
 BEGIN
 IF :Y = 0
 BEGIN
 SET Y (1)
 END
 IF :Y > 4
 BEIN
 HALT
 END
 SET :X (Hello)
 PRINT :X
 SET :Y (:Y + 1)
 PROCEDURE recursion
 END
- c. PROCEDURE recursion
 BEGIN
 LOOP 4
 BEGIN
 SET :X (Hello)
 PRINT :X
 END
 END
- d. PROCEDURE recursion
 BEGIN
 SET :X (Hello)
 SET :Y (1)
 PRINT :X
 PROCEDURE recursion
 IF :Y = 4
 BEGIN
 HALT
 END
 END
- e. PROCEDURE recursion
 BEGIN
 SET :X (Hello)
 PRINT :X
 PROCEDURE recursion
 END

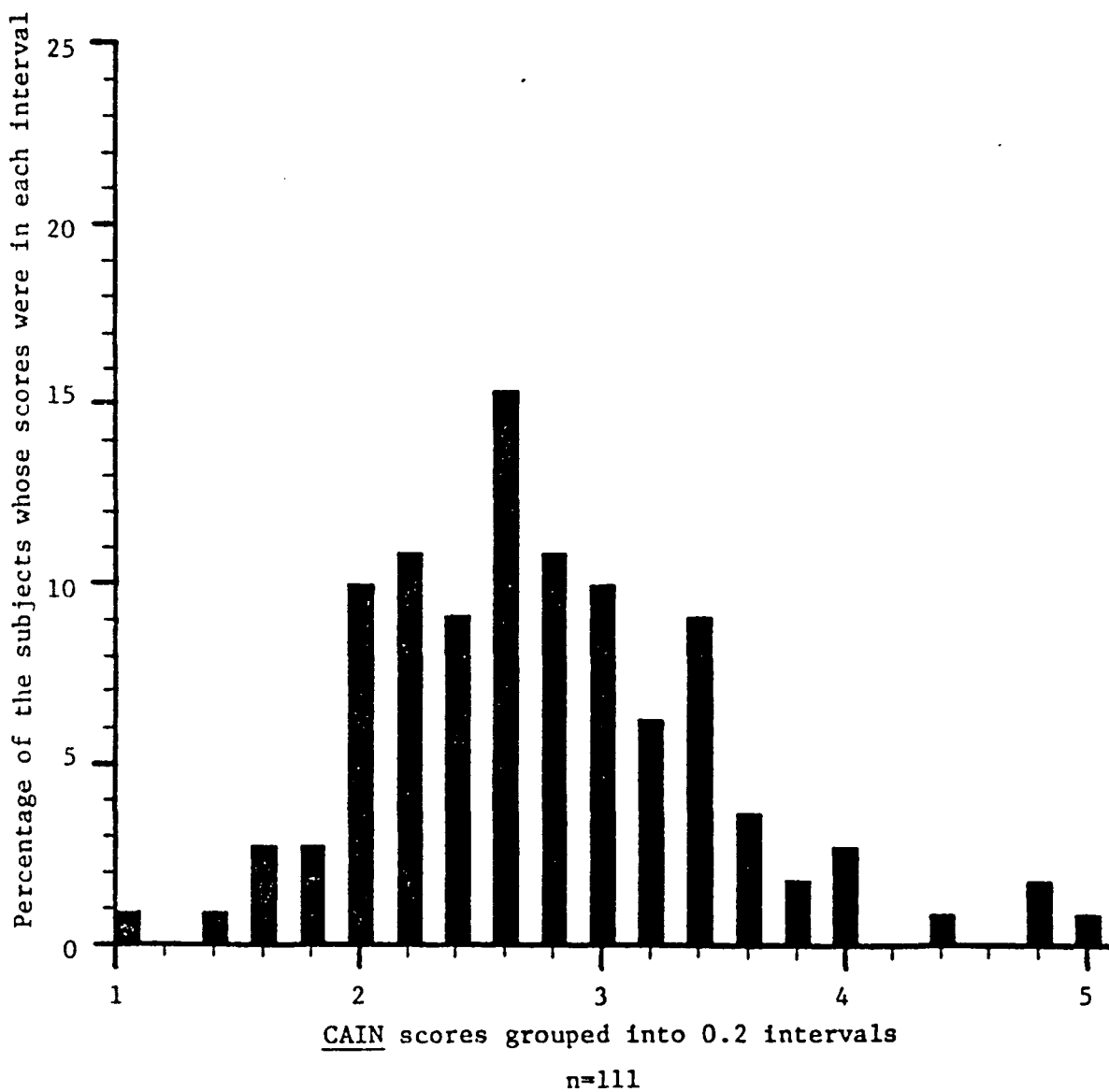
APPENDIX G:

NORMATIVE DATA FROM MAURER'S
COMPUTER ANXIETY INDEX (1983)

Table 9. Means, standard deviations, and ranges of CAIN scores by norm groups (Maurer, 1983, p. 46)

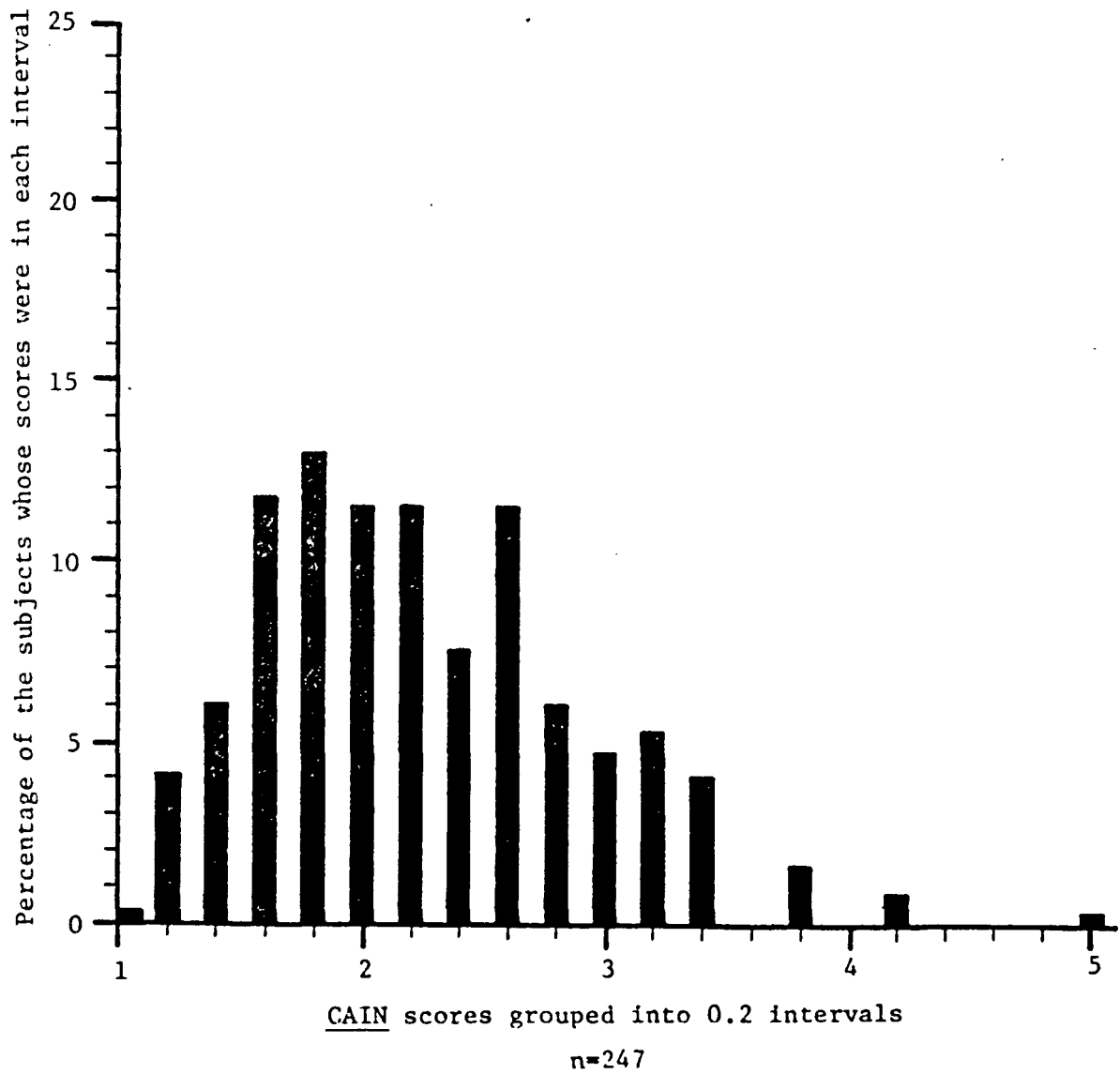
	College student	Junior high	Teacher	Professional	User	Other	All subjects
N	111	247	42	67	122	25	614
Mean score	2.70 ^a	2.21	2.44	1.78	1.99	2.21	2.23
Standard deviation	0.71	0.67	0.92	0.58	0.54	0.72	0.72
Low score (1=lowest possible)	1.00	1.00	1.00	1.00	1.00	1.12	1.00
High score (6=highest possible)	5.04	5.04	4.69	3.73	4.28	4.31	5.04

^aThe higher the score, the more computer anxious the individual.



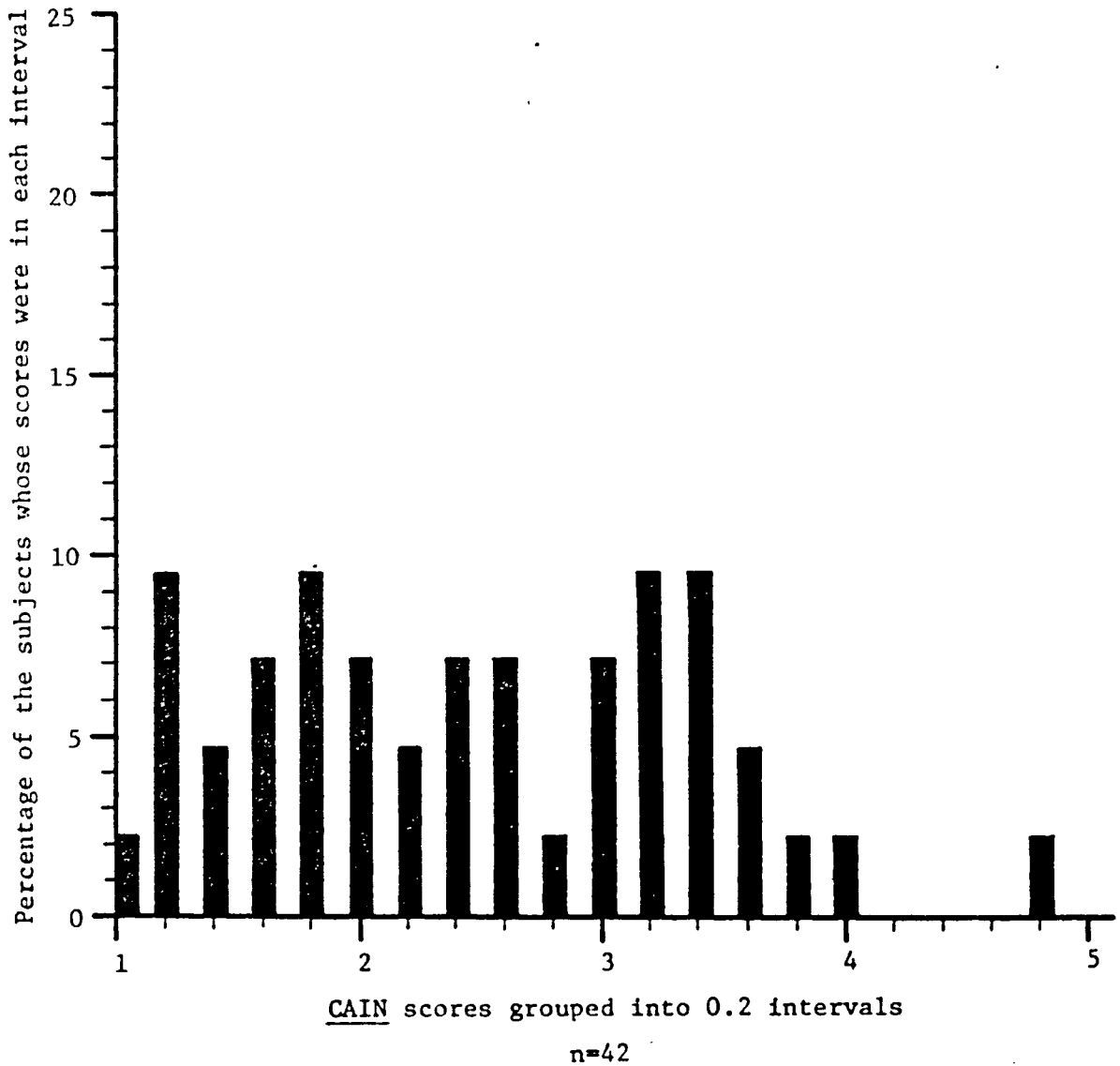
The grouped CAIN scores are reported as a percentage of the total group to show distribution of scores and to allow comparison between groups.

Figure 6. CAIN scores of college students (Maurer, 1983, p. 47)



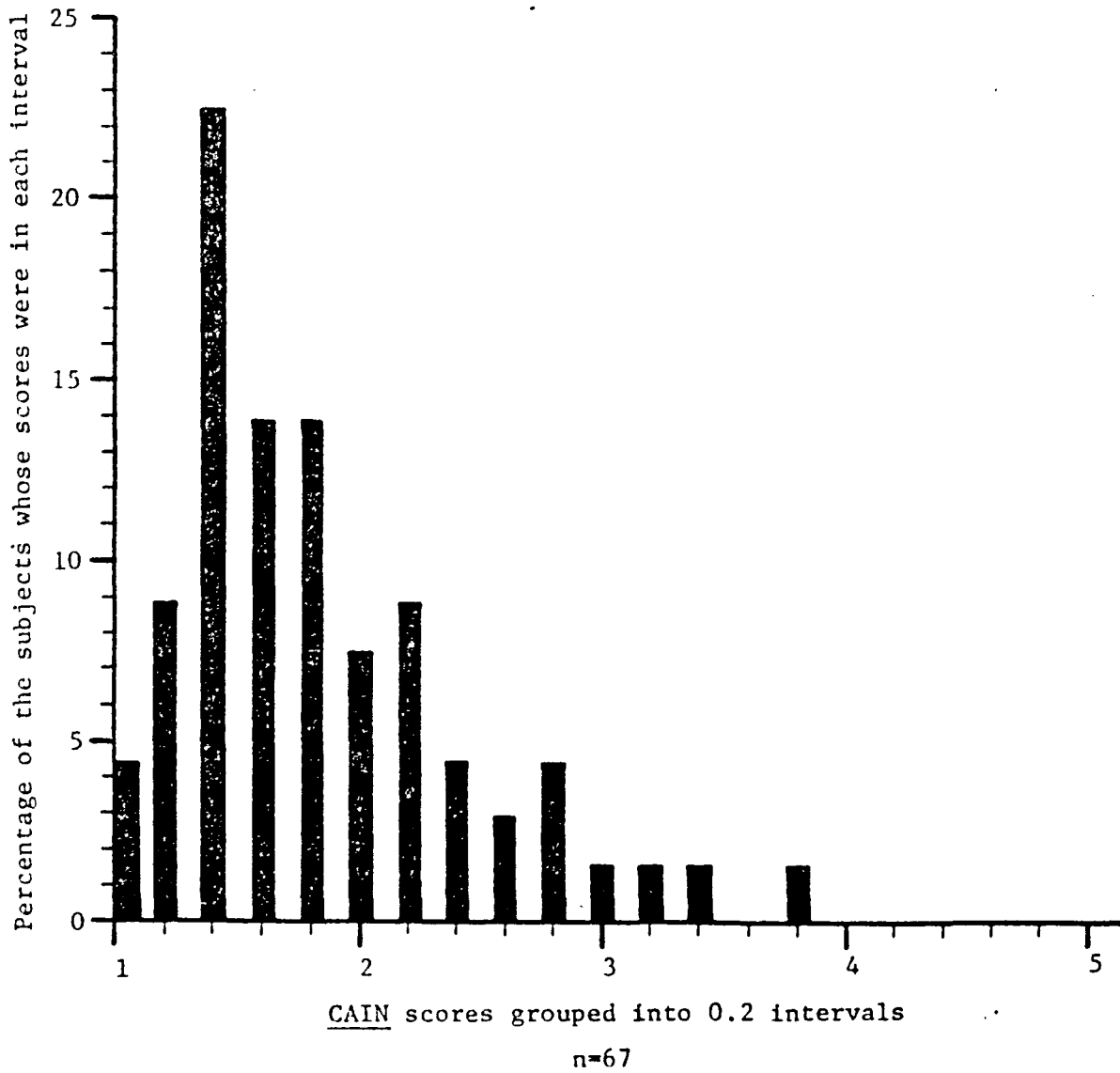
The grouped CAIN scores are reported as a percentage of the total group to show distribution of scores and to allow comparison between groups.

Figure 7. CAIN scores of junior high school students (Maurer, 1983, p. 48)



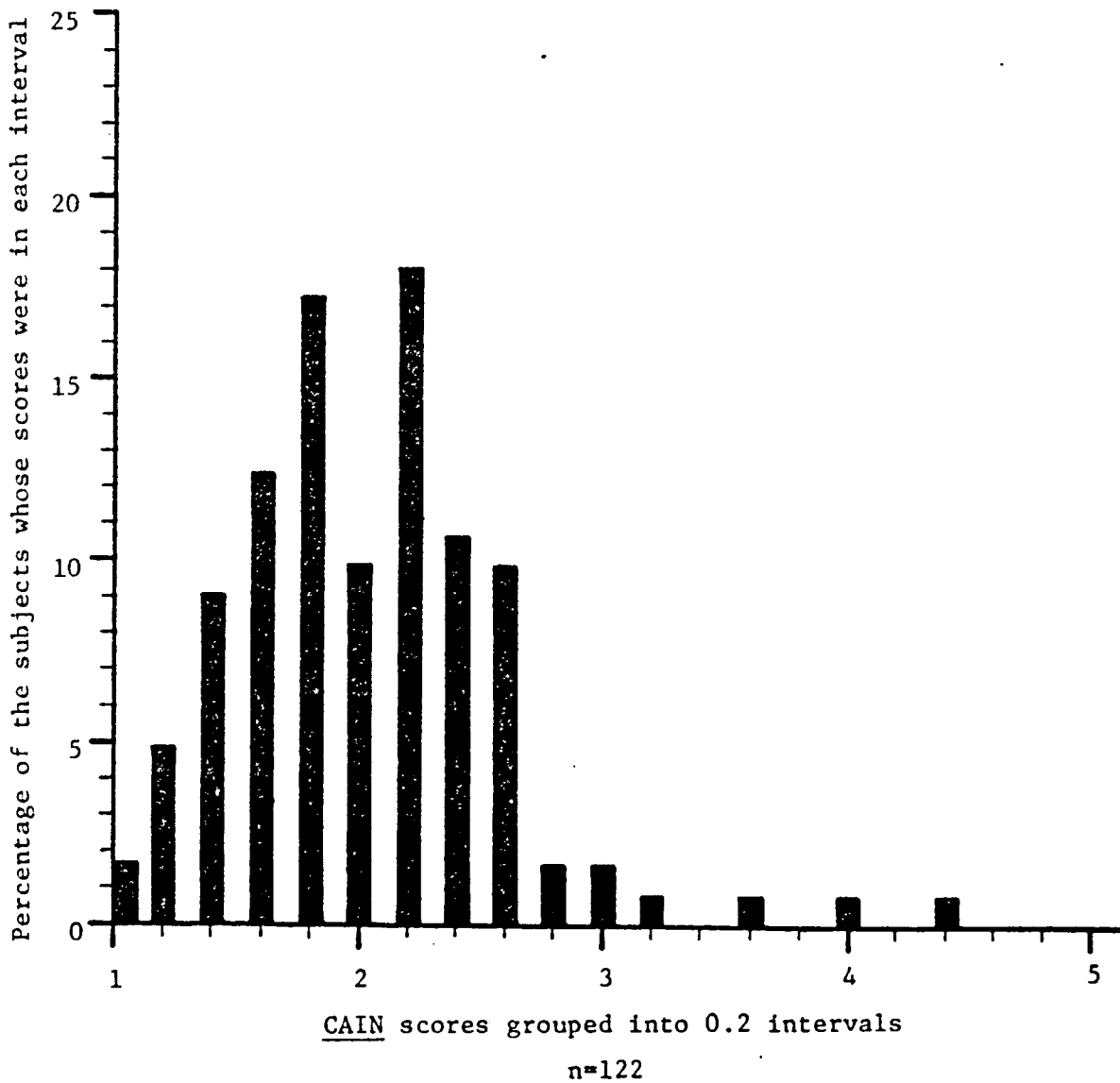
The grouped CAIN scores are reported as a percentage of the total group to show distribution of scores and to allow comparison between groups.

Figure 8. CAIN scores of teachers (Maurer, 1983, p. 49)



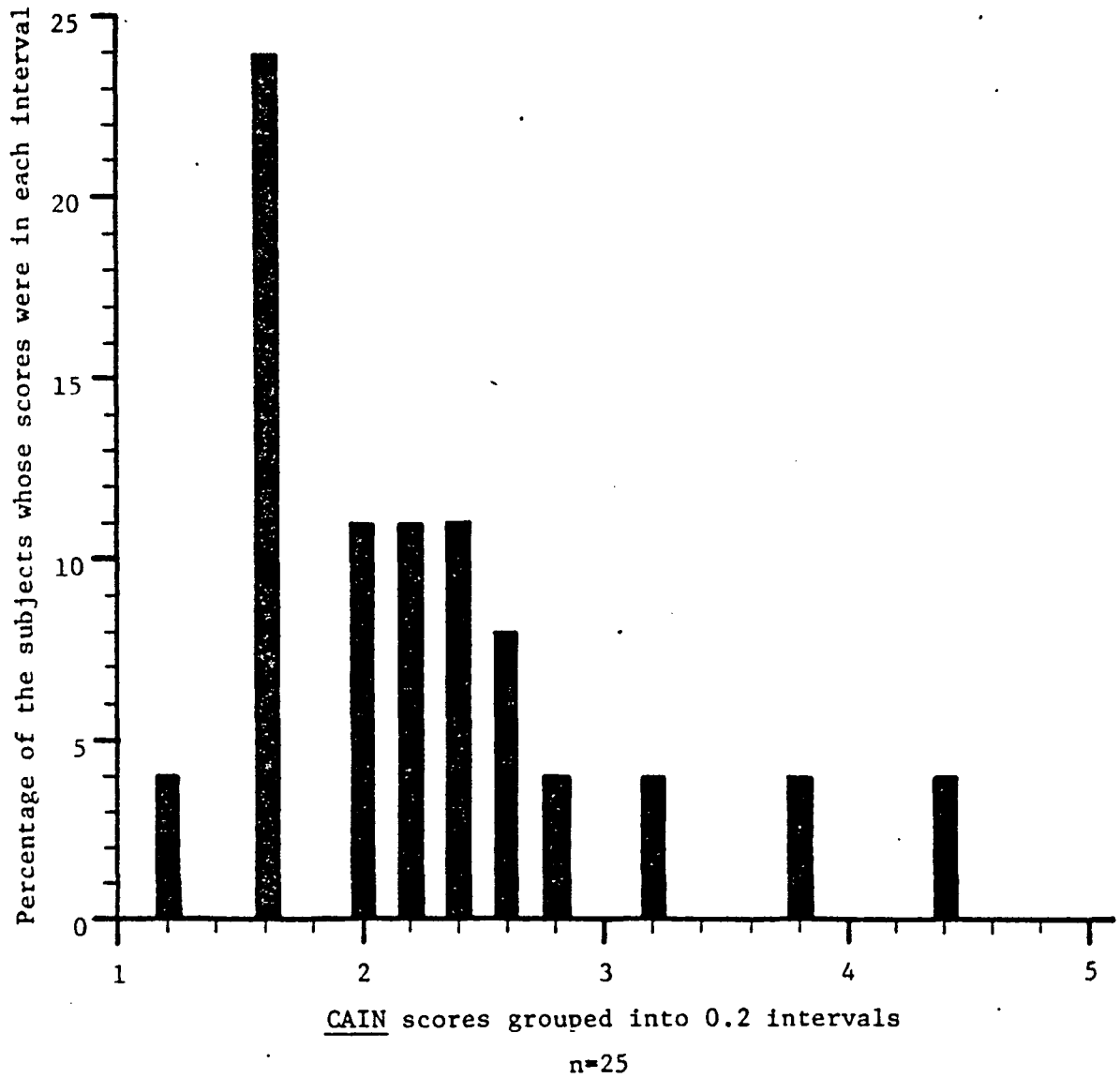
The grouped CAIN scores are reported as a percentage of the total group to show distribution of scores and to allow comparison between groups.

Figure 9. CAIN scores of data processing professionals (Maurer, 1983, p. 50)



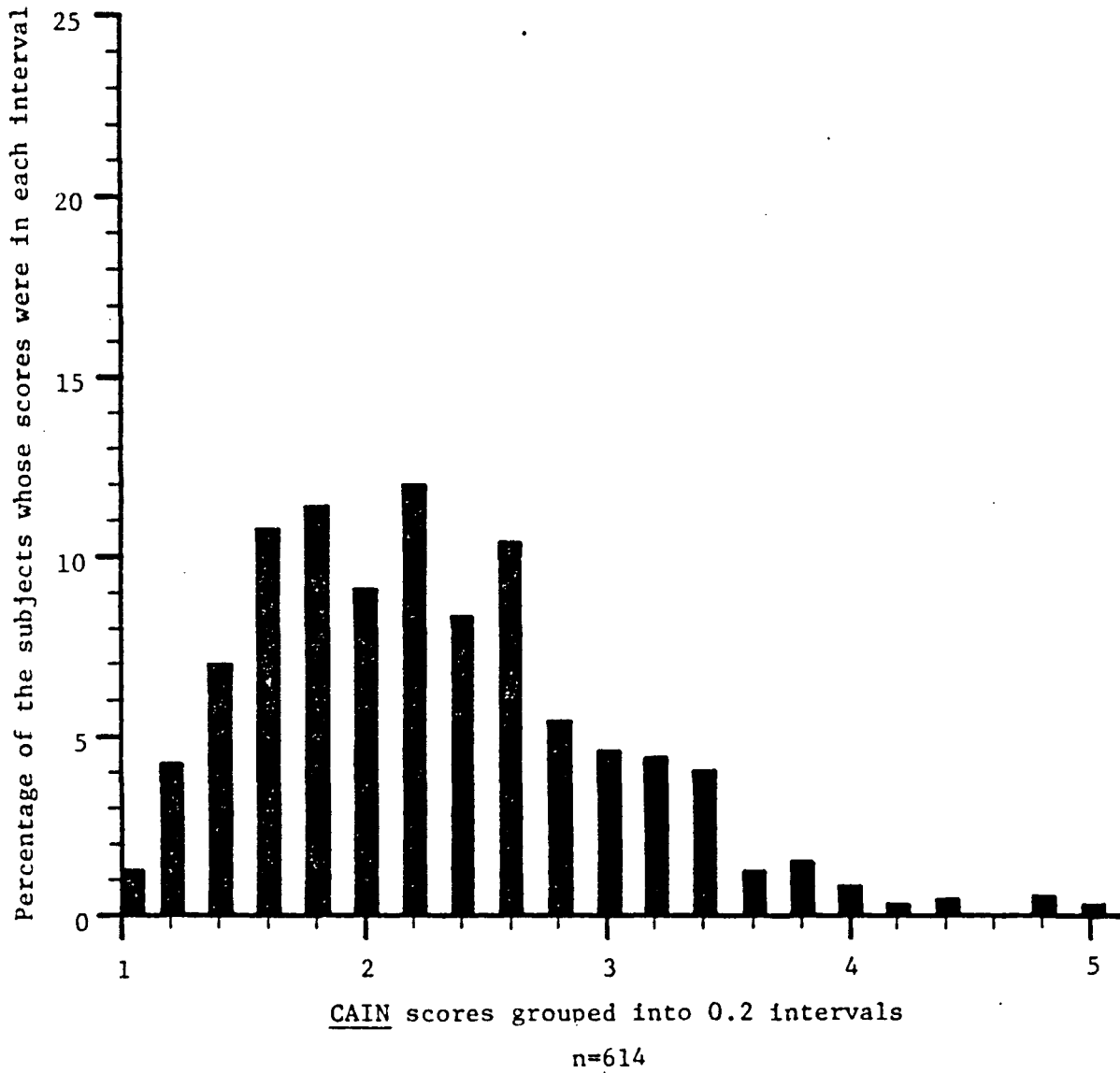
The grouped CAIN scores are reported as a percentage of the total group to show distribution of scores and to allow comparison between groups.

Figure 10. CAIN scores of users of data processing services (Maurer, 1983, p. 51)



The grouped CAIN scores are reported as a percentage of the total group to show distribution of scores and to allow comparison between groups.

Figure 11. CAIN scores of individuals not belonging to any previous group (Maurer, 1983, p. 52)



The grouped CAIN scores are reported as a percentage of the total group to show distribution of scores and to allow comparison between groups.

Figure 12. CAIN scores of all groups (Maurer, 1983, p. 53)

APPENDIX H:
HUMAN SUBJECTS APPROVAL FORM
AND CONSENT FORMS

PILOT TEST FOR A COMPUTER LITERACY ASSESSMENT INSTRUMENT

To: Students enrolled in Secondary Education 301, Secondary Education 102, and Secondary Education 302 and Curriculum and Instructional Media 505 during Spring Semester, 1984

Re: Participation as subjects in a pilot test conducted to obtain data about multiple choice test items written for a criterion-referenced computer literacy assessment instrument being developed as part of a Master's Degree thesis.

From: Mary Montag, graduate student in Curriculum and Instructional Media, Professional Studies in Education

As undergraduate and/or graduate students at Iowa State University who may or may not be instructed in computer literacy, you are being asked to participate in the pilot testing of multiple choice test items written for a criterion-referenced computer literacy assessment instrument. Your participation in this study will involve answering the following multiple choice test items to the best of your ability. Participation in this pilot test is voluntary and consenting or declining to participate will not effect your grade in any course. The participants will not be identified by name and all responses will be kept confidential. The results will be reported in statistical terms only. Any questions or concerns regarding your participation in this study should be referred to Mary Montag, N062 Quadrangle, 294-9305.

PURPOSE OF THE COMPUTER LITERACY ASSESSMENT INSTRUMENT

The computer literacy assessment instrument will attempt to measure the computer literacy knowledge and skills of students at Iowa State University prior to and after taking college-level computer literacy courses. The results of the test may be used in the following ways:

1. To document the pre-college computing competencies of students at Iowa State University and develop normative data about various subgroups of students.

2. To justify revisions in the content of introductory computing courses to reflect student needs and prerequisite skills.
3. To provide criteria for the placement of students in appropriate computing courses based on their level of computer literacy.

PURPOSE OF THE PILOT TEST

The pilot testing phase of the development of a criterion-referenced test involves the administration of test items written from clearly defined domain specifications to two samples of subjects: students instructed in the domain and students uninstructed in the domain measured by the test. In this study, the test items will be administered to students who have received instruction and students who have not received instruction in the domain of computer literacy. A statistical item analysis of the results of the two groups will be conducted to identify the items that will be included in the final test, the items that need to be revised prior to inclusion in the final test, and the items that will be not be included in the final test. The item analysis will help to establish the content validity of the instrument.

If you consent to participate in this pilot test, please follow the directions on the first page of the test booklet to fill out the answer sheet and answer the test items. Thank you very much for your participation and cooperation in this study.

ADMINISTRATION OF A COMPUTER LITERACY ASSESEMENT INSTRUMENT

To: Students enrolled in Secondary Education 301 and
Secondary Education 101 during Spring Semester, 1984

Re: Participation as subjects in the administration of a
criterion-referenced computer literacy assessment
instrument developed as part of a Master's Degree
thesis.

From: Mary Montag, graduate student in Curriculum and
Instructional Media, Professional Studies in
Education

As undergraduate students at Iowa State University who may or may not be instructed in computer literacy, you are being asked to participate in the administration of a criterion-referenced computer literacy assessment instrument to determine the reliability and validity of the instrument, establish a performance standard, and assess normative data based on the characteristics of the subjects. Your participation in this study will involve answering the following multiple choice test items to the best of your ability. Participation in the study is voluntary and consenting or declining to participate will not effect your grade in any course. The participants will not be identified by name and all responses will be kept confidential. The results will be reported in statistical terms only. Any questions or concerns regarding your participation in this study should be referred to Mary Montag, N062 Quadrangle, 294-9305.

PURPOSE OF THE COMPUTER LITERACY ASSESSMENT INSTRUMENT

The computer literacy assessment instrument will attempt to measure the computer literacy knowledge and skills of undergraduate students at Iowa State University prior to taking college-level computer literacy courses. The results of the test may be used in the following ways:

1. To document the pre-college computing competencies of students at Iowa State University and develop normative data about various subgroups of students.

2. To justify revisions in the content of introductory computing courses to reflect student needs and prerequisite skills.
3. To provide criteria for the placement of students in appropriate computing courses based on their level of computer literacy.

EXPLANATION OF THIS ADMINISTRATION OF THE INSTRUMENT

The determination of reliability and validity, and the establishment of a performance standard for a criterion referenced test involves the administration of the test to two samples of subjects: students instructed in the domain and students uninstructed in the domain measured by the test. In this study, the test items will be administered to students who have received instruction and students who have not received instruction in the domain of computer literacy. A statistical analysis of the results of the two groups will be conducted to establish reliability, validity, and the performance standard.

If you consent to participate in this study, please follow the directions on the first page of the test booklet to fill out the answer sheet and answer the test items. Thank you very much for your participation and cooperation in this study.