MICHIGAN TEST FOR TEACHER CERTIFICATION (MTTC)

TEST OBJECTIVES
FIELD 050: COMPUTER SCIENCE

Subarea	Approximate Percentage of Questions on Test
Educational Computing and Technology Literacy	22%
Computer Systems, Data, and Algorithms	21%
Program Design and Verification	19%
Programming Language Concepts	19%
Professional Preparation	19%

I. EDUCATIONAL COMPUTING AND TECHNOLOGY LITERACY

001 Understand basic computer technology operations and concepts.

Includes appropriate use of terminology related to computer technology; characteristics and functions of basic components of computer systems, including hardware, software, and peripherals; basic procedures related to the use of computers (e.g., saving, moving, and backing up files; inputting and retrieving data); features of computer networks; the historical development of and important trends affecting the evolution of computers and related technology; and basic knowledge of the uses of computer technology in business, industry, and society.

002 Understand the selection, installation, management, and maintenance of computer/technology systems and software.

Includes the evaluation and selection of computer/technology systems and software for a given purpose; the configuration of computer/technology systems and related peripherals (e.g., scanners, digital cameras, video cameras); procedures for the organization, management, and security of hardware and software; and strategies for troubleshooting and maintaining various hardware/software configurations.

003 Understand the use of educational and productivity software.

Includes the use of utilities and word processing, desktop publishing, and graphics programs for professional applications; the use of spreadsheets for analyzing, organizing, and graphically displaying data; procedures for designing and manipulating databases and creating customized reports; the use of teacher utility and classroom management tools to design solutions for specific purposes; selection and evaluation of educational software; and features of applications that integrate word processing, database, spreadsheet, communication, and other tools.

Copyright © 2005 by National Evaluation Systems, Inc. (NES®)
"NES®" and its logo are registered trademarks of National Evaluation Systems, Inc.™

This document may not be reproduced for commercial use but may be copied for educational purposes.

004 Understand the use of telecommunications and information access resources.

Includes types, characteristics, and uses of telecommunications tools and resources, including distance learning; the use of electronic mail, Web browser applications, and Web-page creation tools for communication and for research to support instruction; and the use of automated online search tools and intelligent agents to identify and index desired information resources.

005 Understand the use of computers and other technologies in research, education, problem solving, and product development.

Includes basic principles of instructional design associated with the development of multimedia and hypermedia learning materials; the development of multimedia and hypermedia products; the selection of appropriate tools for communicating concepts, conducting research, and solving problems for an intended audience and purpose; the use of technology to support oral presentations; procedures for designing and publishing simple online documents; methods of conducting research and evaluating online sources of information; and strategies and tools for working on collaborative projects, team activities, and online workgroups.

II. COMPUTER SYSTEMS, DATA, AND ALGORITHMS

006 Understand basic characteristics of computer architecture.

Includes characteristics and functions of computer components (e.g., CPU, memory, mass storage devices); machine-level data representation; data storage; the transfer of data from one location to another; steps in the machine cycle and their synchronization; and factors limiting the accuracy of numerical computations.

007 Understand characteristics and functions of operating systems.

Includes the importance of an operating system in coordinating a computer's activities; comparison of single-user and multiuser systems on various platforms; the principles and processes of time sharing; characteristics of and methods for managing computer components and processes (e.g., devices, memory, virtual memory, files, multitasking) through operating systems; and characteristics of the client-server model and the relationship of operating systems to this model.

008 Understand types and characteristics of computer networks.

Includes characteristics of local area networks (LANs) and wide area networks (WANs); various network configurations and their characteristics; the basic structure and features of the Internet; basic principles of data transfer on the Internet and the role of network protocols; and security issues related to networks and the Internet.

Copyright © 2005 by National Evaluation Systems, Inc. (NES®)

This document may not be reproduced for commercial use but may be copied for educational purposes.

009 Understand types and applications of data structures.

Includes characteristics and uses of arrays, stacks, queues, linked lists, and binary trees; the function of pointers (e.g., in relation to dynamic data structures); characteristics of abstract data types; and principles of encapsulation and its role in maintaining data integrity.

010 Understand characteristics and uses of algorithms.

Includes the role of algorithms in computing; general characteristics of algorithms (e.g., correctness, efficiency, finiteness); standard algorithms (e.g., searching, sorting); problem-solving approaches; methods used to design and represent algorithms (e.g., pseudocode); characteristics and uses of iterative and recursive structures; methods of testing algorithms; and the analysis of algorithms (e.g., time-and-space trade-offs, big-O notation).

III. PROGRAM DESIGN AND VERIFICATION

011 Apply principles and procedures for designing and preparing a program.

Includes the process of analyzing a problem or application to be addressed by a program; the role of design specifications; the uses of flowcharts, schematic drawings, and pseudocode in program preparation; differences in top-down and bottom-up design methodologies; the function of modules in program design; modularization strategies; and principles of inheritance in object-oriented programming.

012 Apply program development and implementation procedures.

Includes steps in the programming process; characteristics of robust programs and considerations in developing robust programs with effective user interfaces; the use of libraries in programming; the purposes of programming style conventions (e.g., indenting, spacing, comments) and their appropriate application; object-based development strategies; strategies for modifying existing programs; software tools for developing programs; and steps in the process of program implementation (i.e., translation, linking, loading).

013 Apply program verification principles and procedures.

Includes the purpose of program testing; the design and evaluation of a test plan; common programming errors; procedures for locating errors; and methods for debugging programs.

014 Apply documentation and communication principles to software development.

Includes characteristics and purposes of user and system documentation of programs; the creation of clear and appropriate program documentation; the role of written and oral communication in team-based software development projects; and the application of communication skills in computer science–related contexts.

Copyright © 2005 by National Evaluation Systems, Inc. (NES®)

This document may not be reproduced for commercial use but may be copied for educational purposes.

IV. PROGRAMMING LANGUAGE CONCEPTS

015 Understand types and characteristics of programming languages.

Includes differences in levels of programming language (e.g., machine language); an overview of the historical development of programming languages; functions of compilers and interpreters; characteristics of various programming paradigms (e.g., imperative, functional, object-oriented); features of various types of high-level languages; and strengths and limitations of different types of languages for a given application.

016 Understand the characteristics and application of data types and declarations in high-level languages.

Includes characteristics and uses of variables and constants; properties and uses of data types (e.g., integer, character, Boolean); the characteristics and use of declarations to specify constants, variables, data types and structures, functions, and parameters; and the characteristics and uses of inheritance/ classes.

017 Understand types and characteristics of statements, operators, and control structures in high-level languages.

Includes purposes of statements, operators, and control structures; types and characteristics of operators; principles of operator precedence; the role and use of assignment statements; the purpose and use of comments; and characteristics of sequential, conditional, and repetitive control structures.

018 Understand the characteristics and use of program modularization in high-level languages.

Includes characteristics of program modules (e.g., subprograms, objects); the difference between local and global variables; the role of parameters; parameter-passing techniques; methods of executing a module from within a program; characteristics of event-driven software; and implementation of input and output procedures.

Copyright © 2005 by National Evaluation Systems, Inc. (NES®)

This document may not be reproduced for commercial use but may be copied for educational purposes.

V. PROFESSIONAL PREPARATION

019 Understand appropriate materials, methods, resources, and curricula for teaching computer science.

Includes the identification and modeling of problem-solving strategies for computer science instruction; the uses of computers and related technologies as teaching tools for computer science instruction; the selection and use of appropriate materials and models for teaching computer science; the identification of resources to enrich the teaching of computer science; the design, development, and evaluation of laboratory activities and demonstrations for the computer science classroom; and common features of K–12 computer science curricula and their relationship to the college computer science curriculum.

020 Apply practices that reflect the roles and responsibilities of computer science teachers.

Includes guidance roles (e.g., career guidance, college preparation); appropriate enrichment activities for students (e.g., extracurricular computer clubs, organized competitions); and awareness of professional computer science and computer-education associations (e.g., Michigan Association for Computer-Related Technology Users in Learning [MACUL], Association for Computing Machinery [ACM], International Society for Technology in Education [ISTE]).

021 Apply classroom and instructional management methodologies for teaching computer science.

Includes the selection and use of appropriate materials and methods for teaching given computer science content; computer-science laboratory management skills and techniques; the development and implementation of instructional strategies to address the needs of all students (e.g., students with various learning styles, students with special needs); the selection and application of appropriate methods of assessment and evaluation; and appropriate techniques for providing students with feedback.

022 Understand societal issues in computer science.

Includes ethical and legal issues related to the use of computer/technology resources (e.g., acceptable use, privacy, copyright, security); the role of computers and related technology resources in promoting lifelong learning; the effects of computers and technology on individuals and society; equity issues related to technology (e.g., access, training opportunities); knowledge of gender, cultural, and ethnic issues in computer science; and resources for adaptive assistive devices for students with special needs.

Copyright © 2005 by National Evaluation Systems, Inc. (NES®)

This document may not be reproduced for commercial use but may be copied for educational purposes.