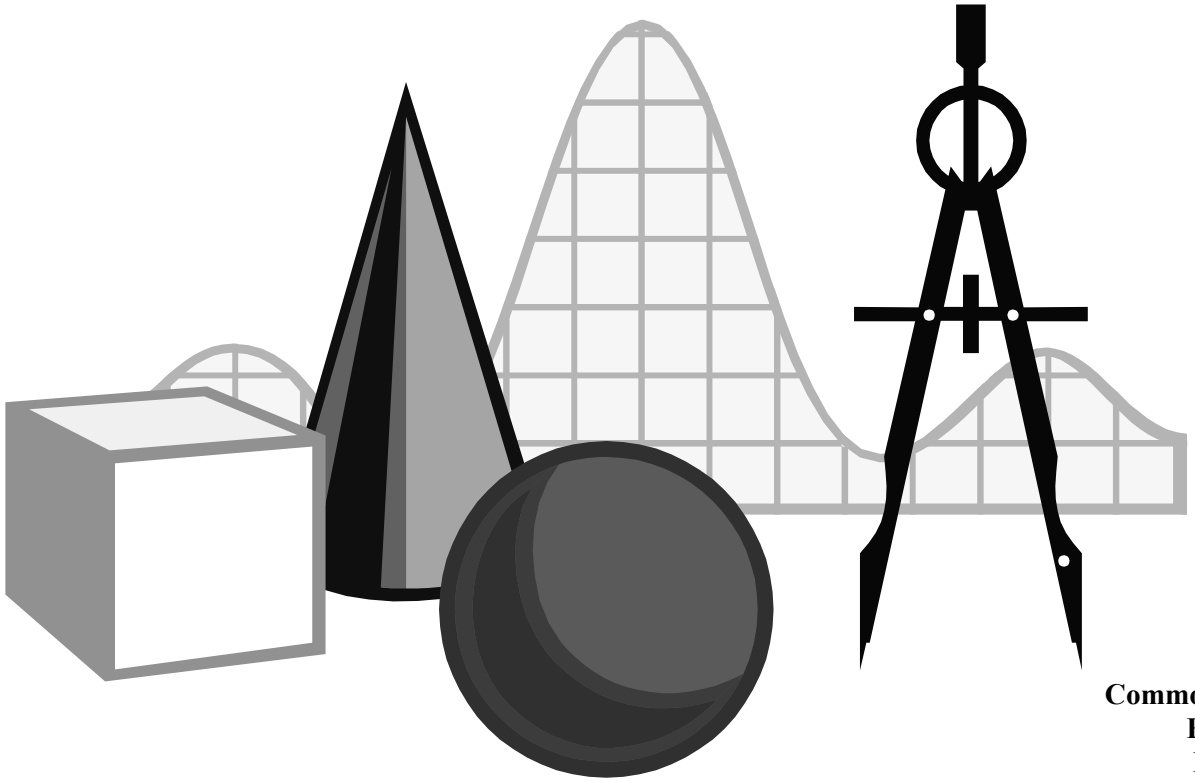
MATHEMATICS STANDARDS OF LEARNING CURRICULUM FRAMEWORK

Computer Mathematics



Commonwealth of Virginia
Board of Education
Richmond, Virginia

Virginia Department of Education
P.O. Box 2120
Richmond, Virginia 23218-2120
http://www.pen.k12.va.us

Superintendent of Public Instruction
Jo Lynne DeMary

Deputy Superintendent
M. Kenneth Magill

Assistant Superintendent for Instruction
Patricia I. Wright

Office of Secondary Instructional Services
Linda M. Wallinger, Director
Deborah Kiger Lyman, Mathematics Specialist

NOTICE TO THE READER

The Virginia Department of Education does not unlawfully discriminate on the basis of sex, race, color, religion, handicapping conditions, or national origin in employment or in its educational programs and activities.

The 2002 *Mathematics Curriculum Framework* can be found in PDF and Microsoft Word file formats on the Virginia Department of Education's website at http://www.pen.k12.va.us.

Introduction

Mathematics content develops sequentially in concert with a set of processes that are common to different bodies of mathematics knowledge. The content of the Mathematics Standards of Learning supports five process goals for students: becoming mathematical problem solvers, communicating mathematically, reasoning mathematically, making mathematical connections, and using mathematical representations to model and interpret practical situations. These goals provide a context within which to develop the knowledge and skills identified in the standards.

Computer Mathematics provides students with experiences in using the computer to solve problems that can be set up as mathematical models. Students will develop and refine skills in logic, organization, and precise expression, thereby enhancing learning in other disciplines. Programming should be introduced in the context of mathematical concepts and problem solving. Students will define a problem; develop, refine, and implement a plan; and test and revise the solution.

Each topic in the Computer Mathematics Curriculum Framework is developed around the Standards of Learning. Each Standard of Learning is expanded in the Essential Knowledge and Skills column. The Essential Understandings column includes concepts, mathematical relationships, and ideas that are important to understanding and teaching the Standard of Learning effectively.

Teachers should help students make connections and build relationships among algebra, arithmetic, geometry, discrete mathematics, and probability and statistics. Connections should be made to other subject areas and fields of endeavor through applications. Using manipulatives, graphing calculators, and computer applications to develop concepts should help students develop and attach meaning to abstract ideas. Throughout the study of mathematics, students should be encouraged to talk about mathematics, use the language and symbols of mathematics, communicate, discuss problems and problem solving, and develop their competence and their confidence in themselves as mathematics students.

TOPIC: PROBLEM SOLVING

COMPUTER MATHEMATICS
STANDARD COM.1

The student will apply programming techniques and skills to solve practical problems in mathematics arising from consumer, business, personal finance, leisure activities, sports, probability and statistics, and other applications in mathematics. Problems will include opportunities for students to analyze data in charts, graphs, and tables and to use their knowledge of equations, formulas, and functions to solve these problems.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• The computer is an essential tool for mathematical problem solving in consumer-related problems. • A subtask that has been solved previously may be used again and again. • Programming languages require the use of particular structures to express algorithms as programs. • Designing algorithms is the problem-solving phase of computer programming. • Practical problems that can be modeled mathematically can be solved with a computer program. • Data arising from probability and statistics applications can be displayed in tables and graphs and analyzed within the structure of a computer program.	• Design and implement computer programs to solve consumer problems. • Analyze and interpret graphs, charts, and tables in the design and implementation of a computer program. • Design and implement computer programs to – solve mathematical problems, using formulas; – solve mathematical problems, using equations; – solve mathematical problems, using functions; – solve problems related to geometry, business, and leisure; – solve probability problems; – solve data-analysis problems; and – solve statistical problems.

TOPIC: PROGRAM DESIGN

COMPUTER MATHEMATICS
STANDARD COM.2

The student will design, write, test, debug, and document a program. Programming documentation will include pre-conditions and post-conditions of program segments, input/output specifications, the step-by-step plan, the test data, a sample run, and the program listing with appropriately placed comments.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• The successful completion of a structured program requires problem-solving skills.	• Describe what a computer program is. • List and describe the stages involved in writing a computer program. • Describe the function of an algorithm. • Describe the interplay between hardware and software in program execution. • Compare and contrast compiling and executing a program. • Determine what a given output statement will print. • Debug a program. • Provide required documentation for a program.

TOPIC: PROGRAM DESIGN

COMPUTER MATHEMATICS
STANDARD COM.3

The student will write program specifications that define the constraints of a given problem. These specifications will include descriptions of pre-conditions, post-conditions, the desired output, analysis of the available input, and an indication as to whether or not the problem is solvable under the given conditions.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• A programmer begins the programming process by analyzing the problem and developing a general solution (algorithm). • The successful completion of a structured program requires problem-solving skills.	• For a given problem, describe the pre-conditions, post-conditions, and desired output. • Determine whether or not a problem is solvable. • Write program specifications that define the constraints of a problem.

TOPIC: PROGRAM DESIGN

COMPUTER MATHEMATICS
STANDARD COM.4

The student will design a step-by-step plan (algorithm) to solve a given problem. The plan will be in the form of a program flowchart, pseudo code, hierarchy chart, and/or data-flow diagram.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• All programs are implementations of algorithms.	• Design a step-by-step plan to solve a problem. • Utilize the following problem-solving formats: – flowchart; – pseudo code; – hierarchy chart; and – data-flow diagram.

TOPIC: PROGRAM DESIGN

COMPUTER MATHEMATICS
STANDARD COM.5

The student will divide a given problem into manageable sections (modules) by task and implement the solution. The modules will include an appropriate user-defined function, subroutines, and procedures. Enrichment topics might include user-defined libraries (units) and object-oriented programming.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• Functional decomposition is a way to develop a program in which the problem is divided into subproblems whose solutions comprise the solution to the original problem.	• Subdivide a problem into modules by task. • Implement the solution of the problem. • Write task-oriented modules, including – a user-defined function; – subroutines; and – procedures. • Determine the need for a subroutine or user-defined function. • Determine the difference between and the need for internal and external subroutines and functions.

TOPIC: PROGRAM DESIGN

COMPUTER MATHEMATICS
STANDARD COM.12

The student will translate a mathematical expression into a computer statement, which involves writing assignment statements and using the order of operations.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• A variable identifies a location in memory where a data value that can be changed is stored. • An assignment statement stores the value of an expression into a variable. • The order of operations is – parentheses; – exponents; – multiplication and division in order from left to right; and – addition and subtraction in order from left to right. • Variable assignment statements will differ depending upon the programming language used.	• Translate a mathematical expression into a computer statement. • Use the order of operations to simplify expressions. • Write variable assignment statements. • Construct and evaluate expressions that include multiple arithmetic operations.

TOPIC: PROGRAM DESIGN

COMPUTER MATHEMATICS
STANDARD COM.13

The student will select and implement built-in (library) functions in processing data.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• The argument of a library function is a value or expression associated with the independent variable. • A library function is a subroutine.	• Use library functions in designing programs to process data. • Use library functions that are arithmetic or string operators. • Invoke a value-returning function.

TOPIC: PROGRAM DESIGN

COMPUTER MATHEMATICS
STANDARD COM.14

The student will implement conditional statements that include "if/then" statements, "if/then/else" statements, case statements, and Boolean logic.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• Boolean logic is a system using variables with only two values: TRUE and FALSE. • The "if" statement is the fundamental control structure that allows branches in the flow of control.	• Construct a simple logical (Boolean) expression to evaluate a given condition. • Construct an "if/then" statement to perform a specific task. • Construct an "if/then/else" statement to perform a specific task. • Use conditional statements to incorporate decision making into programs.

TOPIC: PROGRAM DESIGN

COMPUTER MATHEMATICS
STANDARD COM.17

The student will implement pre-existing algorithms, including sort routines, search routines, and simple animation routines.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• Arranging values into an order is known as sorting. • A sequential search algorithm starts at the beginning of a list and examines each data value in sequence. • Implementation of animation routines will differ depending upon the programming language used.	• Implement pre-existing algorithms into a program. • Implement a sort routine on a one-dimensional array. • Implement a sequential search routine on a one-dimensional array. • Implement a binary search routine on a one-dimensional array. • Implement a simple animation routine.

TOPIC: PROGRAM IMPLEMENTATION

COMPUTER MATHEMATICS
STANDARD COM.6

The student will design and implement the input phase of a program, which will include designing screen layout and getting information into the program by way of user interaction, data statements, and/or file input. The input phase also will include methods of filtering out invalid data (error trapping).

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• A program needs data on which to operate. • A file is a named area in secondary storage that holds a collection of information.	• Design a screen layout to facilitate input. • Design program information input by – user interaction; – data statements (BASIC); and – file input. • Filter out invalid data, using a variety of methods (error trapping). • Construct input statements to read values into a program. • Determine the contents of variables that have been assigned values by input statements.

TOPIC: PROGRAM IMPLEMENTATION

COMPUTER MATHEMATICS
STANDARD COM.7

The student will design and implement the output phase of a computer program, which will include designing output layout, accessing a variety of output devices, using output statements, and labeling results.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• Output is dependent on input. • Implementation of the output portion of a program includes designing the output and displaying it in the desired format.	• Design an output layout. • Access various output devices. • Use output statements. • Label results.

TOPIC: PROGRAM IMPLEMENTATION

COMPUTER MATHEMATICS
STANDARD COM.8

The student will design and implement computer graphics, which will include topics appropriate for the available programming environment as well as student background. Students will use graphics as an end in itself, as an enhancement to other output, and as a vehicle for reinforcing programming techniques.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• Work with computer graphics is specific to the computer operating system.	• Design computer graphics. • Implement computer graphics. • Plot points and areas. • Determine and set window or screen dimensions. • Determine and set screen and background colors. • Use box commands. • Describe the role of graphics in the computer environment.

TOPIC: PROGRAM IMPLEMENTATION

COMPUTER MATHEMATICS
STANDARD COM.15

The student will implement loops, including iterative loops. Other topics will include single entry point, single exit point, pre-conditions, and post-conditions.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• A loop executes a sequence of statements repeatedly. • Nested loops contain other loops.	• Determine when a loop is needed in a program. • Implement loops into programs. Include – iterative loops; – pre-test loops; and – post-test loops. • Incorporate single entry point, single exit point, pre-conditions, and post-conditions into loops.

TOPIC: PROGRAM IMPLEMENTATION

COMPUTER MATHEMATICS
STANDARD COM.16

The student will select and implement appropriate data structures, including arrays (one-dimensional and/or multidimensional), files, and records. Implementation will include creating the data structure, putting information into the structure, and retrieving information from the structure.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• Structured data types are collections of components that are given a single name and whose organization is characterized by the method used to access the individual components. • Multidimensional arrays may be viewed as arrays of one-dimensional arrays.	• Declare a one-dimensional or two-dimensional array for a given problem. • Choose an appropriate component type for an array. • Assign a value to an array component. • Fill an array with data, and process the data in the array. • Access a particular component of a two-dimensional array. • Process a two-dimensional array by rows and by columns. • Retrieve data from an array.

TOPIC: DATA MANIPULATION

COMPUTER MATHEMATICS
STANDARD COM.9

The student will define simple variable data types that include integer, real (fixed and scientific notation), character, string, and Boolean.

STANDARD COM.10

The student will use appropriate variable data types, including integer, real (fixed and scientific notation), character, string, and Boolean. This will also include variables representing structured data types.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• A data type is a set of values and a set of operations on the values. • Boolean data has only two literal constants, and they represent TRUE and FALSE. • A string is an array for which there exists an aggregate constant.	• Define and use variable data types, including – integer; – real (fixed and scientific notation); – character; – string; and – Boolean. • Write numeric and string variables, using valid names.

TOPIC: DATA MANIPULATION

COMPUTER MATHEMATICS
STANDARD COM.11

The student will describe the way the computer stores, accesses, and processes variables, including the following topics: the use of variables versus constants, variables addresses, pointers, parameter passing, scope of variables, and local versus global variables.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• Computers consist of hardware components that interact with software.	• Determine when use of a variable is appropriate. • Describe how a computer stores, accesses, and processes variables. • Incorporate variable addresses, pointers, and parameter passing into programs. • Differentiate between local and global variables, and describe their appropriate use. • Compare and contrast variables and constants.

TOPIC: PROGRAM TESTING

COMPUTER MATHEMATICS
STANDARD COM.18

The student will test a program, using an appropriate set of data. The set of test data should be appropriate and complete for the type of program being tested.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• A program test can reveal problems (bugs) in the program. • Testing a program for bugs is part of problem solving. • Various forms of data can be used to debug a program.	• Test a program, using an appropriate and complete set of data.

TOPIC: PROGRAM TESTING

COMPUTER MATHEMATICS
STANDARD COM.19

The student will debug a program, using appropriate techniques (e.g., appropriately placed controlled breaks, the printing of intermediate results, and other debugging tools available in the programming environment), and identify the difference between syntax errors and logic errors.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• Debugging a program is problem solving.	• Debug a program, using controlled breaks, the printing of intermediate results, and other debugging tools. • Identify the differences among syntax errors, runtime errors, and logic errors.

TOPIC: PROGRAM TESTING

COMPUTER MATHEMATICS
STANDARD COM.20

The student will design, write, test, debug, and document a complete structured program that requires the synthesis of many of the concepts contained in previous standards.

ESSENTIAL UNDERSTANDINGS	ESSENTIAL KNOWLEDGE AND SKILLS
• The successful completion of a structured program requires problem-solving skills.	• Design, write, test, debug, and document a complete structured program.