

**Computer Math Pacing Guide**

	<b>Standards of Learning and Essential Knowledge and Skills</b>	<b>Anticipated Timeframe/week</b>	<b>Teacher Resources and Notes</b>	<b>Assessment Method (optional)</b>
	<p><b><u>PROGRAM DESIGN:</u></b></p> <p>SOLs covered: COM.1, COM.2, COM.3, COM.4, COM.5, COM.18, COM.19, COM.20</p> <ul style="list-style-type: none"> <li>• A programmer begins the programming process by analyzing the problem and developing a general solution (algorithm).</li> <li>• The successful completion of a structured program requires problem solving skills.</li> <li>• All programs are implementations of algorithms.</li> </ul> <p><b>The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to</b></p> <ul style="list-style-type: none"> <li>• For a given problem, describe the preconditions, postconditions, and desired output.</li> <li>• Determine whether or not a problem is solvable.</li> <li>• Write program specifications that define the constraints of a</li> </ul>	<p>Week 1-3</p> <p>Days 1 - 10</p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc">http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc</a></p> <p>Pages 10-14</p>	

## Computer Math Pacing Guide

	<p>problem.</p> <ul style="list-style-type: none"><li>• Design a step-by-step plan to solve a problem.</li><li>• Utilize the following problem solving formats:<ul style="list-style-type: none"><li>– flowchart;</li><li>– pseudo code;</li><li>– hierarchy chart; and</li><li>– data-flow diagram.</li></ul></li></ul>			
--	--	--	--	--

## Computer Math Pacing Guide

<p><b><u>CREATING A PROGRAM:</u></b></p> <p>SOLs covered: COM.12, COM.13, COM.14, COM.15</p> <ul style="list-style-type: none"><li>• The successful completion of a structured program requires problem solving skills.</li></ul> <p><b>The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to</b></p> <ul style="list-style-type: none"><li>• Describe what a computer program is.</li><li>• List and describe the stages involved in writing a computer program.</li><li>• Describe the function of an algorithm.</li><li>• Describe the interplay between hardware and software in program execution.</li><li>• Compare and contrast compiling and executing a program.</li><li>• Determine what a given output statement will print.</li></ul>	<p>Week 3-4</p> <p><b>Days 1 - 13</b></p> <p>Days 11 - 18</p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc">http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc</a></p> <p>Pages 15-16</p>	
---	---	---	--

### Computer Math Pacing Guide

	<ul style="list-style-type: none"><li>• Debug a program.</li><li>• Provide required documentation for a program.</li></ul>			
--	--	--	--	--

**Computer Math Pacing Guide**

	<p><b><u>FORMULAS</u></b></p> <p>SOLs covered: COM.1, COM.2, COM.3, COM.4, COM.5, COM.6, COM.7, COM.9, COM.10, COM.12, COM.13, COM.18, COM.20</p> <ul style="list-style-type: none"> <li>• A variable identifies a location in memory where a data value that can be changed is stored.</li> <li>• An assignment statement stores the value of an expression into a variable.</li> <li>• The order of operations is             <ul style="list-style-type: none"> <li>– parentheses;</li> <li>– exponents;</li> <li>– multiplication and division in order from left to right; and</li> <li>– addition and subtraction in order from left to right.</li> </ul> </li> <li>• Variable assignment statements will differ depending upon the programming language used.</li> </ul> <p><b>The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to</b></p> <ul style="list-style-type: none"> <li>• Translate a mathematical expression into a computer</li> </ul>	<p>Week 4-5</p> <p>Days 14 - 23</p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc">http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc</a></p> <p>Pages 20-26</p>	
--	---	-------------------------------------	---	--

## Computer Math Pacing Guide

	<p>statement.</p> <ul style="list-style-type: none"><li>• Use the order of operations to simplify expressions.</li><li>• Write variable assignment statements.</li><li>• Construct and evaluate expressions that include multiple arithmetic operations.</li></ul>			
--	--	--	--	--

## Computer Math Pacing Guide

<p><b><u>INPUT AND OUTPUT</u></b></p> <p>SOLs covered: COM.6, COM.7, COM.9, COM.10, COM.11, COM.12</p> <ul style="list-style-type: none"><li>• A program needs data on which to operate.</li><li>• A file is a named area in secondary storage that holds a collection of information.</li><li>• Output is dependent on input.</li><li>• Implementation of the output portion of a program includes designing the output and displaying it in the desired format.</li></ul> <p><b>The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to</b></p> <ul style="list-style-type: none"><li>• Design a screen layout to facilitate input.</li><li>• Design program information input by<ul style="list-style-type: none"><li>– user interaction;</li><li>– data statements (BASIC);</li><li>and</li></ul></li></ul>	<p>Week 6-7</p> <p>Days 24 - 32</p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc">http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc</a></p> <p>Pages 17-19</p>	
---	-------------------------------------	---	--

**Computer Math Pacing Guide**

	<ul style="list-style-type: none"><li>- file input.</li><li>• Filter out invalid data, using a variety of methods (error trapping).</li><li>• Construct input statements to read values into a program.</li><li>• Determine the contents of variables that have been assigned values by input statements.</li><li>• Design an output layout.</li><li>• Access various output devices.</li><li>• Use output statements.</li><li>• Label results.</li></ul>			
--	---	--	--	--



**Computer Math Pacing Guide**

<p><b><u>CONDITIONALS</u></b></p> <p><u>SOLs covered:</u> COM.1, COM.2, COM.3, COM.4, COM.5, COM.6, COM.7, COM.9, COM.10, COM.12, COM.13, COM.14, COM.18, COM.20</p> <ul style="list-style-type: none"> <li>• Boolean logic is a system using variables with only two values: TRUE and FALSE.</li> <li>• The “if” statement is the fundamental control structure that allows branches in the flow of control.</li> </ul> <p><b>The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to</b></p> <ul style="list-style-type: none"> <li>• Construct a simple logical (Boolean) expression to evaluate a given condition.</li> <li>• Construct an “if/then” statement to perform a specific task.</li> <li>• Construct an “if/then/else” statement to perform a specific task.</li> <li>• Use conditional statements to incorporate decision making</li> </ul>	<p>Week 7-9</p> <p>Days 28 - 42</p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc">http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc</a></p> <p>Pages 27-35</p>	
---	-------------------------------------	---	--

**Computer Math Pacing Guide**

	into programs.			
--	----------------	--	--	--

## Computer Math Pacing Guide

<p><b><u>LOOPS</u></b></p> <p>SOLs covered: 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.</p> <ul style="list-style-type: none"><li>• A loop executes a sequence of statements repeatedly.</li><li>• Nested loops contain other loops.</li></ul> <p><b>The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to</b></p> <ul style="list-style-type: none"><li>• Determine when a loop is needed in a program.</li><li>• Implement loops into programs. Include<ul style="list-style-type: none"><li>– iterative loops;</li><li>– pretest loops; and</li><li>– posttest loops.</li></ul></li><li>• Incorporate single entry point, single exit point, pre-conditions, and post-conditions into loops.</li></ul>	<p>Week 10-12 Days 43 - 54</p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc">http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc</a></p> <p>Pages 36-45</p>	
---	------------------------------------	---	--

## Computer Math Pacing Guide

<p><b><u>OCCUPATIONAL AND CONSUMER APPLICATIONS</u></b></p> <p>SOLs covered: COM.1, COM.2, COM.3, COM.4, COM.5, COM.6, COM.7, COM.9, COM.10, COM.11, COM.12, COM.13, COM.14, COM.15, COM.16, COM.17, COM.18, COM.19, COM.20</p> <ul style="list-style-type: none"><li>• The computer is an essential tool for mathematical problem solving in consumer-related problems.</li><li>• A subtask that has been solved previously may be used again and again.</li><li>• Programming languages require the use of particular structures to express algorithms as programs.</li><li>• Designing algorithms is the problem solving phase of computer programming.</li><li>• Real-world problems that can be modeled mathematically can be solved with a computer program.</li><li>• Data arising from probability and statistics applications can be displayed in tables and graphs</li></ul>	<p>Week 12-14 Days 55 - 64</p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc">http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc</a></p> <p>Pages 46-56</p>	
--	------------------------------------	---	--

## Computer Math Pacing Guide

and analyzed within the structure of a computer program.

**The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to**

- 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.

<p><b><u>BUILT-IN APPLICATIONS</u></b></p> <p>SOLs covered: COM.1, COM.13</p> <ul style="list-style-type: none"> <li>• The argument of a library function is a value or expression associated with the independent variable.</li> <li>• A library function is a subroutine.</li> </ul> <p><b>The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to</b></p> <ul style="list-style-type: none"> <li>• Use library functions in designing programs to process data.</li> <li>• Use library functions that are arithmetic or string operators.</li> <li>• Invoke a value-returning function.</li> </ul>	<p>Week 14-16</p> <p><b>Days 65 - 74</b></p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc">http://www.doe.virginia.gov/instruction/high_school/mathematics/computer_math_graphcalc.doc</a></p> <p>Pages 57-58</p>	
---	--	---	--

## Computer Math Pacing Guide

<p><b><u>SUBROUTINES</u></b></p> <p>SOLs covered: COM.1, COM.2, COM.3, COM.4, COM.5, COM.6, COM.7, COM.10, COM.11, COM.12, COM.14, COM.15, COM.18, COM.19, COM.20</p> <ul style="list-style-type: none"><li>• A library function is a subroutine.</li><li>• 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.</li></ul> <p><b>The student will use problem solving, mathematical communication, mathematical reasoning, connections, and representations to</b></p> <ul style="list-style-type: none"><li>• Subdivide a problem into modules by task.</li><li>• Implement the solution of the problem.</li><li>• Write task-oriented modules, including<ul style="list-style-type: none"><li>– a user-defined function;</li><li>– subroutines; and</li><li>– procedures.</li></ul></li><li>• Determine the need for a subroutine or user-defined</li></ul>	<p>Week 16-17 Days 75 - 83</p>	<p><a href="http://www.doe.virginia.gov/instruction/high_school/math/computer_math_graphcalc.do">http://www.doe.virginia.gov/instruction/high_school/math/computer_math_graphcalc.do</a></p> <p>Pages 59-74</p>	
---	------------------------------------	---	--

## Computer Math Pacing Guide

	<p>function.</p> <ul style="list-style-type: none"><li>• Determine the difference between and the need for internal and external subroutines and functions</li></ul>			
--	--	--	--	--