

## HCMS Computer Science Curriculum

UNIT	STANDARDS	LEARNING CONCEPTS
Keyboarding Technique	<p>1.1 Technique: Demonstrate proper typing technique, including correct hand and finger placement, maintaining an appropriate posture, and using all fingers for typing. Emphasize the importance of using the home row as a base and minimizing reliance on looking at the keyboard while typing.</p>	<ul style="list-style-type: none"> <li>● Exhibits proper posture and fingering techniques for the alphanumeric keyboard.</li> <li>● Applies the touch-keying system to develop basic skills on the alphanumeric keyboard at a rate of 15 gross words a minute (GWAM) for a one-minute timed writing.</li> <li>● Practices proper response patterns to gain speed and accuracy.</li> <li>● Uses the correct touch system to reach the numeric and symbol keys.</li> </ul>
Foundational Concepts	<p>MS.FC: Foundational Concepts</p> <p>1) Analyze the advantages and limitations of existing computing devices to improve user experience.</p> <p>2) Demonstrate skills in identifying and solving hardware and software problems that can occur during regular usage.</p> <p>3) Apply computational thinking to a variety of problems across multiple disciplines.</p> <p>4) Understand how collaboration is essential to computer science and apply collaborative skills to develop computational solutions.</p>	<ul style="list-style-type: none"> <li>● Computing systems' hardware</li> <li>● Computing systems' software</li> <li>● Investigable questions/problems that can be solved by computational thinking/CS</li> <li>● Collaboration about computer science</li> </ul>
ALGORITHMIC THINKING	<p>MS.AT: Algorithmic Thinking</p> <p>1) Use clearly named variables of various data types to create generalized algorithms.</p> <p>2) Create algorithms which</p>	<ul style="list-style-type: none"> <li>● Logical or sequential thinking</li> <li>● Breaking down problems into parts</li> <li>● Create step by step</li> </ul>

	<p>include methods of controlling the flow of computation using “if...then... else” type conditional statements to perform different operations depending on the values of inputs.</p> <p>3) Identify algorithms that make use of sequencing, selection, or iteration.</p> <p>4) Describe how algorithmic processes and automation increase efficiency.</p>	<p>processes to solve problems</p>
<p>Programming Concepts</p>	<p>1) Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.</p> <p>2) Create procedures with parameters that hide the complexity of a task and can be reused to solve similar tasks.</p> <p>3) Seek and incorporate feedback from team members and users to refine a solution that meets user needs.</p> <p>4) Provide proper attribution when incorporating existing code, media, and libraries into original programs.</p> <p>5) Use the iterative design process to systematically test and refine programs to improve performance and eliminate errors.</p> <p>6) Document programs using comments and/or README files to make them easier to follow, test, and debug.</p> <p>7) Design a function using a programming language.</p>	<ul style="list-style-type: none"> <li>● Using and troubleshooting existing code</li> <li>● Develop a code to solve a problem</li> <li>● Concepts of programming(language exposure)</li> </ul>
<p>Data Analysis</p>	<p>1) Represent data using multiple encoding schemes, such as decimal, binary, Unicode, Morse code, Shorthand,</p>	<ul style="list-style-type: none"> <li>● Organizing, clustering or categorizing data</li> <li>● Cleaning or transforming data to discover useful</li> </ul>

	<p>student-created codes.</p> <p>2) Refine computational models based on the data they have generated.</p> <p>3) Collect, analyze, transform, and refine computational data to make it more useful and reliable.</p>	<p>information</p> <ul style="list-style-type: none"> <li>● Displaying or visualizing data</li> <li>● Communicating data for decision making or problem solving</li> </ul>
<p>Networking and the Internet</p>	<p>1) Identify and employ appropriate troubleshooting techniques used to solve computing or connectivity issues. 19</p> <p>2) Differentiate between secure and non-secure websites and applications including how they affect and use personal data.</p> <p>3) Describe the causes and effects of intellectual property as it relates to print and digital media, considering copyright, fair use, licensing, sharing, and attribution.</p> <p>4) Compare and contrast common methods of securing data and cybersecurity.</p> <p>5) Analyze different modes of social engineering and their effectiveness.</p>	<ul style="list-style-type: none"> <li>● Connectivity and issues</li> <li>● Data is transferred using protocols and rules</li> <li>● Protecting data and personal information</li> </ul>
<p>Impacts of Computing</p>	<p>1) Identify and evaluate the impacts computer science innovations have had on our society.</p> <p>2) Identify how computational systems are being used to collect and analyze information both public and private and understand the benefits and disadvantages of these systems for the user and developer.</p> <p>3) Cite evidence of the positive and negative effects of data permanence on personal and professional digital identity.</p> <p>4) Discuss digital globalization</p>	<ul style="list-style-type: none"> <li>● Innovations due to computer science</li> <li>● Benefits and risks to users and developers due to computational systems (internet safety)</li> <li>● How jobs/careers use computer science and computational thinking</li> </ul>

	<p>and Internet censorship. 5) Investigate a variety of education pathways and career options that utilize computational thinking and/or computer science skills across the state of Tennessee and the world.</p>	
--	---	--