



## Math Department Curriculum Guide

## Computer Science Principles – Advanced Placement

Course Description	
<p>The Computer Science Principles – Advanced Placement (AP) course is designed to be equivalent to a first-semester introductory college computing course. In this course, students will develop computational thinking vital for success across all disciplines, such as using computational tools to analyze and study data and working with large data sets to analyze, visualize, and draw conclusions from trends. The course is unique in its focus on fostering student creativity. Students are encouraged to apply creative processes when developing computational artifacts and to think creatively while using computer software and other technology to explore questions that interest them. They will also develop effective communication and collaboration skills, working individually and collaboratively to solve problems, and discussing and writing about the importance of these problems and the impacts to their community, society, and the world.</p>	
AP CollegeBoard	
<p>The Computer Science Principles – AP curriculum is defined by the CollegeBoard and is compatible with many first year computing courses in colleges and universities. Any student who enrolls in an AP course is required to take the AP exam in May of the school year. A full description of this course and the corresponding exam can be found <a href="#">on the CollegeBoard website</a>.</p>	
Course Content	Computational Thinking Practices
<p>The Computer Science Principles – AP course is organized into five Big Ideas. The approximate weighting on multiple-choice portion of the AP exam is included.</p> <ul style="list-style-type: none"> <li>■ Big Idea 1: Creative Development (10-13%)</li> <li>■ Big Idea 2: Data (17-22%)</li> <li>■ Big Idea 3: Algorithms and Programming (30-35%)</li> <li>■ Big Idea 4: Computing Systems &amp; Networks (11-15%)</li> <li>■ Big Idea 5: Impact of Computing (21-26%)</li> </ul>	<ul style="list-style-type: none"> <li>■ <b>Computational Solution Design:</b> Design and evaluate computational solutions for a purpose.</li> <li>■ <b>Algorithms and Program Development:</b> Develop and implement algorithms.</li> <li>■ <b>Abstraction in Program Development:</b> Develop programs that incorporate abstractions.</li> <li>■ <b>Code Analysis:</b> Evaluate and test algorithms and programs.</li> <li>■ <b>Computing Innovations:</b> Investigate computing innovations.</li> <li>■ <b>Responsible Computing:</b> Contribute to an inclusive, safe, collaborative, and ethical computing culture.</li> </ul>
Computer Science Resources	
<p>Hanover High School students also use the CodeHS platform to learn computer science skills. With a focus on helping students develop problem solving and computational thinking skills, Computer Science Principles – AP students can expect to come away both with a knowledge of professional programming languages and the conceptual understanding needed to learn new languages.</p>	



**Subject: Computer Science Principles – Advanced Placement**

Units	Content
<b>Unit 1 - Introduction to Programming</b>  Term 1 September	<ul style="list-style-type: none"> <li>▪ Welcome to AP CSP</li> <li>▪ Introduction to Programming with Karel</li> <li>▪ More Basic Karel</li> <li>▪ Karel Can't Turn Right</li> <li>▪ Functions in Karel</li> <li>▪ The start() Function</li> <li>▪ Top Down Design and Decomposition in Karel</li> <li>▪ Commenting Your Code</li> <li>▪ Abstraction</li> <li>▪ Super Karel</li> <li>▪ for Loops</li> <li>▪ if Statements</li> <li>▪ if-else Statements</li> <li>▪ while Loops</li> <li>▪ Control Structures</li> <li>▪ Debugging Strategies</li> <li>▪ Karel Algorithms</li> <li>▪ How to Indent Your Code</li> <li>▪ Ultra Karel</li> <li>▪ Karel Challenges</li> </ul>
<b>Unit 2 - Practice PT: Paint</b>  Term 1 September	<ul style="list-style-type: none"> <li>▪ Practice Performance Task: Paint (Pair-Programming)</li> </ul>
<b>Unit 3 - Programming with Javascript</b>  Term 1 October	<ul style="list-style-type: none"> <li>▪ What is Code</li> <li>▪ Uses of Programs</li> <li>▪ Hello World</li> <li>▪ Variables</li> <li>▪ User Input</li> <li>▪ Basic Math in JavaScript</li> <li>▪ Using Graphics in JavaScript</li> <li>▪ Mouse Events: Mouse Clicked</li> </ul>



Units	Content
<b>Unit 4 - JavaScript Control Structures</b>  Term 1 October	<ul style="list-style-type: none"> <li>▪ Booleans</li> <li>▪ Logical Operators</li> <li>▪ Comparison Operators</li> <li>▪ if Statements</li> <li>▪ Key Events</li> <li>▪ For Loops in JavaScript</li> <li>▪ General for Loops</li> <li>▪ For Loop Practice</li> <li>▪ Random Numbers</li> <li>▪ While Loops</li> <li>▪ Loop and a Half</li> </ul>
<b>Unit 5 - Functions and Parameters</b>  Term 1 November	<ul style="list-style-type: none"> <li>▪ Functions and Parameters</li> <li>▪ Functions and Return Values</li> <li>▪ Local Variables and Scope</li> <li>▪ JavaScript vs. Karel</li> <li>▪ Basic JavaScript and Graphics Challenges</li> </ul>
<b>Unit 6 - Basic Data Structures</b>  Term 2 December	<ul style="list-style-type: none"> <li>▪ Introduction to Arrays</li> <li>▪ Indexing into an Array</li> <li>▪ Adding/Removing from an Array</li> <li>▪ Array Length and Looping through Arrays</li> <li>▪ Iterating Over an Array</li> <li>▪ Finding an Element in an Array</li> <li>▪ Removing an Element from an Array</li> <li>▪ Simulation</li> </ul>
<b>Unit 7 - Animation &amp; Games</b>  Term 2 December	<ul style="list-style-type: none"> <li>▪ Canvas Element Manipulation</li> <li>▪ Animation via Timers</li> <li>▪ Mouse Events: Mouse Move, Mouse Up, Mouse Down, Mouse Drag</li> <li>▪ Project: Breakout Arcade Game</li> </ul>



Units	Content
<b>Unit 8 - Digital Information</b>  Term 2 January	<ul style="list-style-type: none"> <li>▪ Introduction to Digital Information</li> <li>▪ Number Systems</li> <li>▪ Encoding Text with Binary</li> <li>▪ Pixel Images</li> <li>▪ The Hexadecimal System</li> <li>▪ Pixel Colors</li> <li>▪ Image Manipulation</li> <li>▪ Data Compression</li> <li>▪ Lossy Compression</li> <li>▪ Cryptography</li> </ul>
<b>Unit 9 - Practice PT: Image Filter</b>  Term 2 January	<ul style="list-style-type: none"> <li>▪ Practice Performance Task: Image Filter (Pair-Programming)</li> </ul>
<b>Unit 10 - The Internet</b>  Term 3 February	<ul style="list-style-type: none"> <li>▪ Introduction to the Internet</li> <li>▪ Internet Hardware</li> <li>▪ Internet Addresses</li> <li>▪ Viewing Websites</li> <li>▪ DNS and Routing</li> <li>▪ Routing</li> <li>▪ Packets and Protocols</li> <li>▪ Sequential, Parallel, and Distributed Computing</li> <li>▪ Cybersecurity</li> <li>▪ The Impact of the Internet</li> <li>▪ Creative Credit and Copyright</li> </ul>
<b>Unit 11 - Data</b>  Term 3 March	<ul style="list-style-type: none"> <li>▪ Getting Started with Data</li> <li>▪ Visualizing and Interpreting Data</li> <li>▪ Data Collection and Limitations</li> </ul>
<b>Unit 12 - Create Performance Task</b>  Term 3 April	<ul style="list-style-type: none"> <li>▪ Program Code</li> <li>▪ Video Recording</li> <li>▪ Written Responses</li> </ul>
<b>Unit 13 - Independent Research</b> <i>(post exam)</i> Term 4 May/June	<ul style="list-style-type: none"> <li>▪ Project Definition</li> <li>▪ Independent Research</li> <li>▪ Class Presentation</li> </ul>