

**Course/Subject: Fundamentals of Coding** **Grade Level: 5**

**Textbook(s) / Instructional Materials Used:** Online resources through code.org

**Month(s): August - October** **Marking Period 1**

**Unit Name**

<u>Big Idea</u>	<u>CSTA K-12 Computer Science Standards</u>	<u>Essential Questions &amp; Lesson Essential Question</u>	<u>Concepts/Objectives</u>	<u>Vocabulary</u>	<u>Competencies</u>
Event Conditionals Functions Digital Citizenship Copyright and Creativity	<p>1B-AP-08 - Compare and refine multiple algorithms for the same task and determine which is the most appropriate.</p> <p>1B-AP-11 - Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.</p> <p><b><u>ISTE Standards</u></b>  <b>Computational Thinker</b></p> <p>Students develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test solutions. Students:</p>	<p>How can building a set of directions help to complete a task?</p> <p>What strategies can be used when coding to find and fix bugs?</p> <p>In what ways do debugging strategies help in modifying an existing program to solve errors?</p> <p>What does it mean to become frustrated?</p> <p>How do I persevere through a problem and keep working when I become frustrated?</p> <p>How can I create a program using to complete images using sequential steps?</p> <p>When can a loop be used to simplify a repetitive action?</p>	Algorithms Debugging Sequencing Persistence Frustration Program Programming Loop Repeat	Algorithm Bug Debugging Loop Program Repeat Programming Frustrated Persistence	Algorithms Debugging Sequencing Programming Loop

	<p>a. formulate problem definitions suited for technology- assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.</p> <p>b. collect data or identify relevant data sets, use digital tools to analyze them, and represent data in various ways to facilitate problem-solving and decision-making.</p> <p>c. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving.</p> <p>d. understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p>		<ul style="list-style-type: none"> <li>• Order movement commands as sequential steps in a program.</li> <li>• Modify an existing program to solve errors.</li> <li>• Break down a long sequence of instructions into the largest repeatable sequence.</li> <li>• Outline steps to complete a structural engineering challenge.</li> <li>• Predict and discuss potential issues in structure creation.</li> <li>• Build a structure based on team plan.</li> <li>• Revise both the plan and structure until they satisfy the challenge.</li> <li>• Predict where a program will fail.</li> <li>• Modify an existing program to solve errors.</li> </ul>		
--	---	--	--	--	--

Month(s): November - January			Marking Period 2		
Unit Name					
<u>Big Idea</u>	<u>CSTA K-12 Computer Science Standards</u>	<u>Essential Questions &amp; Lesson Essential Question</u>	<u>Concepts/Objectives</u>	<u>Vocabulary</u>	<u>Competencies</u>
Event Conditionals Functions Digital Citizenship Copyright and Creativity	<p>1B-AP-08 - Compare and refine multiple algorithms for the same task and determine which is the most appropriate.</p> <p>1B-AP-11 - Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.</p> <p>1B-AP-13 - Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.</p> <p><b><u>ISTE Standards</u></b>  <b>Computational Thinker</b></p> <p>Students develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test</p>	<p>When can a loop be used to simplify a repetitive action?</p> <p>What purpose do loops serve when writing a large set of code that repeats?</p> <p>How do loops improve existing code that repeats?</p> <p>What is the difference between a loop and a nested loop?</p> <p>When is a loop, nested loop, or no loop needed when writing code?</p> <p>What role do conditionals play when writing code for a program?</p> <p>What are the circumstances when certain parts of a program should run and when they shouldn't?</p> <p>What criteria needs to be met to make something a conditional?</p> <p>What is a function?</p> <p>How can functions make programs easier to write?</p> <p>How can predetermined functions</p>	<p>Loop            Nested Loops            Conditionals            Functions</p> <p>Students will be able to:</p> <ul style="list-style-type: none"> <li>Identify the benefits of using a loop structure instead of manual repetition.</li> <li>Differentiate between commands that need to be repeated in loops and commands that should be used on their own.</li> <li>Break complex tasks into smaller repeatable sections.</li> <li>Recognize large repeated patterns as made from smaller repeated patterns.</li> <li>Identify the benefits of using a loop structure</li> </ul>	<p>Loop            Repeat            Conditionals            Condition            While Loop            Function</p>	<p>Loop            Nested Loops            Conditionals            Functions</p>

	<p>solutions. Students:</p> <ol style="list-style-type: none"> <li>a. formulate problem definitions suited for technology- assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.</li> <li>b. collect data or identify relevant data sets, use digital tools to analyze them, and represent data in various ways to facilitate problem-solving and decision-making.</li> <li>c. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving.</li> <li>d. understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</li> </ol>	<p>be used to complete commonly repeated tasks?</p>	<p>instead of manual repetition.</p> <ul style="list-style-type: none"> <li>• Describe when a loop, nested loop, or no loop is needed.</li> <li>• Recognize the difference between using a loop and a nested loop.</li> <li>• Break apart code into the largest repeatable sequences using both loops and nested loops.</li> <li>• Define circumstances when certain parts of a program should run and when they shouldn't.</li> <li>• Determine whether a conditional is met based on criteria.</li> <li>• Traverse a program and predict the outcome, given a set of input.</li> <li>• Define circumstances when certain parts of a program should run and when they shouldn't.</li> <li>• Determine whether a conditional is met</li> </ul>		
--	--	---	--	--	--

			<p>based on criteria.</p> <ul style="list-style-type: none"> <li>• Use functions to simplify complex programs.</li> <li>• Use predetermined functions to complete commonly repeated tasks.</li> </ul>		
--	--	--	---	--	--

<b>Month(s): January - March</b>	<b>Marking Period 3</b>
----------------------------------	-------------------------

<b>Unit Name</b>
------------------

<u>Big Idea</u>	<u>CSTA K-12 Computer Science Standards</u>	<u>Essential Questions &amp; Lesson Essential Question</u>	<u>Concepts/Objectives</u>	<u>Vocabulary</u>	<u>Competencies</u>
<p>Event Conditionals Functions Digital Citizenship Copyright and Creativity</p>	<p>1B-NI-05 - Discuss real-world cybersecurity problems and how personal information can be protected.</p> <p>1B-AP-09 - Create programs that use variables to store and modify data.</p> <p>1B-AP-11 - Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.</p> <p><b><u>ISTE Standards</u></b> <b>Computational Thinker</b></p>	<p>What is cyberbullying, and how do you deal with it?</p> <p>What is a variable, and how is it used in programming?</p> <p>What are situations in the real-world where variables are used?</p> <p>How can variables be used in place of repetitive values inside a program?</p> <p>How do variables impact loops in a program?</p> <p>What is the difference between a loop and a for loop?</p>	<p>Digital Citizenship Variable For Loops</p> <p>Students will be able to:</p> <ul style="list-style-type: none"> <li>• Empathize with those who have received mean and hurtful messages.</li> <li>• Judge what it means to cross the line from harmless to harmful communication online.</li> <li>• Generate solutions for dealing with cyberbullying.</li> </ul>	<p>Cyberbully Variable Constant For Loop</p>	<p>Variable For Loops</p>

	<p>Students develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test solutions. Students:</p> <ol style="list-style-type: none"> <li>a. formulate problem definitions suited for technology- assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.</li> <li>b. collect data or identify relevant data sets, use digital tools to analyze them, and represent data in various ways to facilitate problem-solving and decision-making.</li> <li>c. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving.</li> <li>d. understand how automation works and</li> </ol>		<ul style="list-style-type: none"> <li>• Identify variables and determine their values.</li> <li>• Define and call variables in the context of real-life activities.</li> <li>• Create situations which require the use of variables.</li> <li>• Assign values to existing variables.</li> <li>• Utilize variables in place of repetitive values inside of a program.</li> <li>• Identify areas where they can use variables to modify quantities during runtime.</li> <li>• Examine code to find places where variables can be substituted for specific values.</li> <li>• Use variables to change values inside of a loop.</li> <li>• Determine starting value, stopping value, and stepping</li> </ul>		
--	---	--	---	--	--

	<p>use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</p> <p><b>Digital Citizen</b> Students recognize the rights, responsibilities and opportunities of living, learning and working in an interconnected digital world, and they act and model in ways that are safe, legal and ethical. Students:</p> <ol style="list-style-type: none"><li>a. cultivate and manage their digital identity and reputation and are aware of the permanence of their actions in the digital world.</li><li>b. engage in positive, safe, legal and ethical behavior when using technology, including social interactions online or when using networked devices.</li><li>c. demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property.</li></ol>		<p>value for a 'for' loop.</p> <ul style="list-style-type: none"><li>• Illustrate the counter values hit each time through a for loop during runtime.</li></ul>		
--	---	--	---	--	--

	d. manage their personal data to maintain digital privacy and security and are aware of data-collection technology used to track their navigation online.				
--	---	--	--	--	--

<b>Month(s): March - June</b>	<b>Marking Period 4</b>
-------------------------------	-------------------------

<b>Unit Name</b>
------------------

<u>Big Idea</u>	<u>CSTA K-12 Computer Science Standards</u>	<u>Essential Questions &amp; Lesson Essential Question</u>	<u>Concepts/Objectives</u>	<u>Vocabulary</u>	<u>Competencies</u>
Event Conditionals Functions Digital Citizenship Copyright and Creativity	<p>1B-AP-09 - Create programs that use variables to store and modify data.</p> <p>1B-AP-10 - Create programs that include sequences, events, loops, and conditionals.</p> <p>1B-AP-11 - Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.</p> <p>1B-AP-12 - Modify, remix or incorporate portions of an existing program into one's work, to develop something new or add more advanced features.</p>	<p>What is the difference between a for loop, a loop and a while loop?</p> <p>When should a for loop be used?</p> <p>What is a sprite?</p> <p>What role does an event play when coding a program?</p> <p>How can I use an event to help create an interactive program?</p> <p>What are the steps in the design process?</p> <p>How can the design process help in creating a computer science project?</p>	<p>For Loop Event</p> <p><b>Objectives:</b></p> <p>Students will be able to:</p> <ul style="list-style-type: none"> <li>Determine starting value, stopping value, and stepping value for a 'for' loop.</li> <li>Recognize when to use a 'for' loop and when to use other loops such as 'repeat' and 'while' loops.</li> <li>Use 'for' loops to change loop several times with different values.</li> <li>Define "sprite" as a</li> </ul>	For Loop Behavior Sprite Event Define Prepare Reflect Try	For Loop Event



	<p>1B-AP-13 - Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.</p> <p>1B-AP-14 - Observe intellectual property rights and give appropriate attribution when creating or remixing programs.</p> <p>1B-AP-16 - Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementations and review stages of program development.</p> <p>1B-AP-17 - Describe choices made during program development using code comments, presentations, and demonstrations.</p> <p><b><u>ISTE Standards</u></b> <b>Computational Thinker</b></p> <p>Students develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test</p>		<p>character or object on the screen that can be moved and changed.</p> <ul style="list-style-type: none"> <li>• Create a new sprite and choose its appearance.</li> <li>• Identify actions that correlate to input events.</li> <li>• Create an animated, interactive game using sequence and events.</li> <li>• Learn to plan in advance for an ongoing assignment.</li> <li>• Be able to explain how system limitations can affect project design.</li> <li>• Describe how compromise can help keep a project on track and inspire creativity.</li> <li>• Shape ideas into reasonable goals and plans.</li> <li>• Recognize any potential obstacles such as time constraints or bugs.</li> </ul>		
--	--	--	---	--	--

	<p>solutions. Students:</p> <ol style="list-style-type: none"><li>a. formulate problem definitions suited for technology- assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions.</li><li>b. collect data or identify relevant data sets, use digital tools to analyze them, and represent data in various ways to facilitate problem-solving and decision-making.</li><li>c. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving.</li><li>d. understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.</li></ol>		<ul style="list-style-type: none"><li>• Use the planned design as a blueprint for creation.</li><li>• Overcome obstacles such as time constraints or bugs.</li><li>• Draft and implement plans to resolve any issues in their code.</li><li>• Articulate the design process and how it helped shape the finished culminating project.</li></ul>		
--	---	--	---	--	--