Southern York County School District

One Warrior at a Time

Course/Subject: Fundamentals of Coding	Grade Level: 1

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

Month(s): August - October	Marking Period 1

Unit Name

Big Idea	CSTA K-12 Computer Science Standards	Essential Questions & Lesson Essential Question	Concepts/Objectives	Vocabulary	Competencies
Sequencing Loops Events Copyright & Creativity Digital Citizenship	1A-AP-11 - Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. 1B-IC-18 - Discuss computing technologies that have changed the world and express how those technologies influence, and are influenced by, cultural practices. **ISTE Standards** **Computational Thinker** Students develop and employ strategies for understanding and solving problems in ways that leverage the power of technological methods to develop and test solutions.	What is an algorithm? What strategies can I use when coding to find and fix bugs? How should I behave when working on a computer or in a computer lab? What information is okay to have in your digital footprint?	Debugging Persistence Frustration Click Drag Drop Digital Citizenship Students will be able to: • Define a list of steps (algorithm) to get a friend from their starting point to their goal. • Translate a list of steps into a series of physical actions. • Identify and fix errors in the execution of an algorithm.	Algorithm Bug Debugging Frustrated Persistence Click Double-Click Drag Drop Digital Footprint	Debugging Click Drag Drop

	Students: a. formulate problem definitions suited for technology- assisted methods such as data analysis, abstract models and algorithmic thinking in exploring and finding solutions. 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. c. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving. d. understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions. **Digital Citizen** Students recognize the rights, responsibilities and		• Model proper computer lab behaviors. • Experiment with standard block-based programming activities such as: clicking, drag and drop, etc. • Understand that being safe when they visit websites is similar to staying safe in real life. • Learn to recognize websites that are safe for them to visit. • Recognize if they should ask an adult they trust before they visit a particular website. • Explore what information is appropriate to be put online.		

	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: a. cultivate and manage their digital identity and reputation and are aware of the permanence of their actions in the digital world. b. engage in positive, safe, legal and ethical behavior when using technology, including social interactions online or when using networked devices. c. demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property. d. manage their personal data to maintain digital privacy and security and are aware of data-collection technology used to track their navigation online.				

Month(s): November - January	**Marking Period 2**	

Big Idea	CSTA K-12 Computer Science Standards	Essential Questions & Lesson Essential Question	Concepts/Objectives	Vocabulary	Competencies
Sequencing Loops Events Copyright & Creativity Digital Citizenship	1A-AP-09 - Model the way programs store and manipulate data by using numbers or other symbols to represent information. 1A-AP-11 - Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. 1A-AP-13 - Give attribution when using the ideas and creations of others while developing programs. **ISTE Standards** **Computational Thinker** 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: a. formulate problem definitions suited for technology- assisted methods such as data analysis, abstract	How can building a set of directions help to complete a task? What strategies can be used when coding to find and fix bugs? How can I build a computer program from a written set of directions. Why is not okay to claim that someone else's work?	Algorithms Debugging Program Programming Digital Citizenship Fair Use Students will be able to: • Attend to precision when creating instructions. • Identify and address bugs or errors in sequenced instructions. • Construct a program by reorganizing sequential movements. • Build a computer program from a set of written instructions. • Choose appropriate debugging practices when solving problems. • Explain why it is not	Algorithm Bug Debugging Program Programming	Algorithms Debugging Programming

	models and algorithmic thinking in exploring and finding solutions. 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. c. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving. d. understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions. **Digital Citizen** 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:		okay to claim that someone else's work is your own. • Create original art for the purpose of empathizing with other creators.		

	a. cultivate and manage their digital identity and reputation and are aware of the permanence of their actions in the digital world. b. engage in positive, safe, legal and ethical behavior when using technology, including social interactions online or when using networked devices. c. demonstrate an understanding of and respect for the rights and obligations of using and sharing intellectual property. d. manage their personal data to maintain digital privacy and security and are aware of data-collection technology used to track their navigation online.				

Month(s): January - March			Marking Period 3		
Unit Name					
Big Idea	**CSTA K-12 Computer Science Standards**	**Essential Questions & Lesson Essential Question**	**Concepts/Objectives**	**Vocabulary**	**Competencies**

Sequencing Loops Events Copyright & Creativity Digital Citizenship	1A-AP-09 - Model the way programs store and manipulate data by using numbers or other symbols to represent information. 1A-AP-10 - Develop programs with sequences and simple loops, to express ideas or address a problem. 1A-AP-11 - Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. 1A-AP-14 - Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops. **ISTE Standards** **Computational Thinker** 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: a. formulate problem definitions suited for technology- assisted methods such as data	How can I sequence commands in a logical order to make a program? What strategies can I use when coding to find and fix bugs? How is a loop used to make writing code easier? What does a loop command do?	Programming Loop Repeat Students will be able to: • Sequence commands in a logical order. • Recognize problems or "bugs" in a program and develop a plan to resolve the issues. • Identify repeated patterns in code that could be replaced with a loop. • Write instructions that use loops to repeat patterns. • Construct a program using structures that repeat areas of code. • Improve existing code by finding areas of repetition and moving them into looping structure. • Identify the benefits of using a loop structure instead of manual repetition.	Algorithm Bug Debugging Program Programming Loop Repeat	Programming Loops

	analysis, abstract models and algorithmic thinking in exploring and finding solutions. 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. c. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving. d. understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.		• Break down a long sequence of instructions into the smallest repeated sequence possible.		

Month(s): March - June			Marking Period 4		

Unit Name					
Big Idea	**CSTA K-12 Computer Science Standards**	**Essential Questions & Lesson Essential Question**	**Concepts/Objectives**	**Vocabulary**	**Competencies**

Sequencing Loops Events Copyright & Creativity Digital Citizenship	1A-AP-09 - Model the way programs store and manipulate data by using numbers or other symbols to represent information. 1A-AP-10 - Develop programs with sequences and simple loops, to express ideas or address a problem. 1A-AP-11 - Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. 1A-AP-14 - Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops. **ISTE Standards** **Computational Thinker** 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: a. formulate problem definitions suited for technology- assisted methods such as data	How is a loop used to make writing code easier? What is an event when looking at code? What is the difference between an event and an action? How can events make programs more interactive?	Loop Event Students will be able to: • Count the number of times an action should be repeated and represent it as a loop. • Decompose a shape into its largest repeatable sequence. • Create a program that draws complex shapes by repeating simple sequences. • Repeat commands given by an instructor. • Recognize actions of the teacher as signals to initiate commands. • Practice differentiating pre-defined actions and event-driven ones. • Identify actions that correlate to input events. • Create an animated,	Loop Event	Loops Events

	analysis, abstract models and algorithmic thinking in exploring and finding solutions. 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. c. break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving. d. understand how automation works and use algorithmic thinking to develop a sequence of steps to create and test automated solutions.		interactive story using sequences and event-handlers. • Share a creative artifact with other students.		