

Computer Programming AC

Unit:	Unit 1—Exploring Programming Languages					
Big Ideas:	The link between computational thinking and programming languages and the social implications each are associated with.					
Unit Essential Questions:	What is the link between learning a programming language and computational thinking for learners in the 21 st Century?					
Concept & Pacing	New Emphasis (Pa Core Standard)	Key Vocabulary	Essential Questions	Mini-Lessons/Activities	Instructional Materials	Assessments
<p>Chapter 1: Introducing Scratch 2.0</p> <p>“Getting to Know Scratch 2.0”</p> <p>“Getting Ready for Scratch 2.0”</p> <p>“Creating Your First Scratch 2.0 Application”</p> <p>“Joining Scratch 2.0’s Global Community”</p> <p>1-2 Weeks</p>	<p>NBEA Standards</p> <hr/> <p><u>Programming and Application Development</u></p> <p>Achievement Standard: Design, develop, test, and implement programs and applications.</p> <p>Level 1- 2 Performance Expectations</p> <ul style="list-style-type: none"> ● Identify and define object-oriented programming terminology ● Demonstrate the ability to code using object-oriented programming <p>Level 3–4 Performance Expectations</p> <ul style="list-style-type: none"> ● Identify and explain programming structures ● Differentiate between source and object code ● Use scripting languages in application development ● Apply design principles to programming tasks ● Develop both procedural and object oriented programs ● Select and incorporate appropriate compiler ● Code common tasks using application development tools ● Test, debug, and document code ● Maintain and reengineer existing code ● Develop programs and applications for a variety of platforms ● Design 3D and gaming environments in relationship to the development of applications <p>PA Standards</p> <p>15.4.12.H: Use programming languages to develop logical thinking and problem solving skills.</p>	<p>BASIC C ++ Costumes Java Remixing Scripts Sounds Sprites/Actors Stage</p>	<p>What are the capabilities, uses, and requirements of Scratch 2.0?</p> <p>What are the benefits of the Scratch 2.0 global community and how does this pertain to business collaboration?</p>	<p>Bell-ringers for each section</p> <p>Kahoot review game for test</p> <p>Reflection Questions to be written in Journal</p>	<p>Textbook Internet and class website MS Word MS PowerPoint</p>	<p>Exploring Scratch Research Project</p> <p>Chapter Test</p>

Computer Programming AC

	<p>15.4.12.I: Compare and contrast programming languages; select most appropriate one to complete a specific task.</p> <p>15.4.12.J: Create a complex computer program to solve a problem.</p>					
<p>Chapter 2: Getting Comfortable with the Scratch 2.0 Development Environment</p> <p>“Getting Comfortable with the Scratch 2.0 Project Editor”</p> <p>“Adding Project Instructions, Notes, and Credits”</p> <p>“Creating New Sprites Using Scratch’s Paint Editor”</p> <p>1-2 Weeks</p>	<p>NBEA Standards</p> <hr/> <p><u>Programming and Application Development</u></p> <p>Achievement Standard: Design, develop, test, and implement programs and applications.</p> <p>Level 1- 2 Performance Expectations</p> <ul style="list-style-type: none"> Identify and define object-oriented programming terminology Demonstrate the ability to code using object-oriented programming <p>Level 3–4 Performance Expectations</p> <ul style="list-style-type: none"> Identify and explain programming structures Differentiate between source and object code Use scripting languages in application development Apply design principles to programming tasks Develop both procedural and object oriented programs Select and incorporate appropriate compiler Code common tasks using application development tools Test, debug, and document code Maintain and reengineer existing code Develop programs and applications for a variety of platforms Design 3D and gaming environments in relationship to the development of applications <p><u>PA Standards</u></p> <p>15.4.12.H: Use programming languages to develop logical thinking and problem solving skills.</p> <p>15.4.12.I: Compare and contrast programming languages; select most appropriate one to complete a specific task.</p>	<p>Backpack Block Help Clear Copy Costume Center Cut Delete Duplicate Edit Fade In Fade Out File Grow Hide Horizontal Flip Import Language Louder My Stuff Paste Redo Reverse Rotate-counterclockwise Rotate-clockwise Scratch Shrink Silence Softer Thumbnails Tips Undo Username</p>	<p>What are the components and the menu bar commands of the Scratch 2.0 Project Editor?</p> <p>Why would adding project instructions, notes, and credits to your project be important?</p> <p>How can a new sprite be created and added to a project?</p>	<p>Bell-ringers for each section</p> <p>Kahoot review game for test</p> <p>Reflection Questions to be written in Journal</p>	<p>Textbook Internet and class website MS Word MS PowerPoint</p>	<p>Scratch User Interface Project</p> <p>Chapter Test</p>

Computer Programming AC

<p>Chapter 3: A Review of the Basic Components of Scratch Projects</p> <p>“Working with Blocks and Stacks”</p> <p>“Six Basic Types of Scratch Blocks”</p> <p>“Keeping an Eye Out with Monitors”</p> <p>“Ten Categories of Scratch Blocks”</p> <p>“Getting Help with Code Blocks”</p> <p>1-2 Weeks</p>	<p>15.4.12.J: Create a complex computer program to solve a problem.</p> <p>NBEA Standards</p> <hr/> <p><u>Programming and Application Development</u></p> <p>Achievement Standard: Design, develop, test, and implement programs and applications.</p> <p>Level 1- 2 Performance Expectations</p> <ul style="list-style-type: none"> Identify and define object-oriented programming terminology Demonstrate the ability to code using object-oriented programming <p>Level 3–4 Performance Expectations</p> <ul style="list-style-type: none"> Identify and explain programming structures Differentiate between source and object code Use scripting languages in application development Apply design principles to programming tasks Develop both procedural and object oriented programs Select and incorporate appropriate compiler Code common tasks using application development tools Test, debug, and document code Maintain and reengineer existing code Develop programs and applications for a variety of platforms Design 3D and gaming environments in relationship to the development of applications <p><u>PA Standards</u></p> <p>15.4.12.H: Use programming languages to develop logical thinking and problem solving skills.</p> <p>15.4.12.I: Compare and contrast programming languages; select most appropriate one to complete a specific task.</p> <p>15.4.12.J: Create a complex computer program to solve a problem.</p>	<p>Arguments Boolean Blocks C Blocks Cap Blocks Control Blocks Cloud Variable Data Blocks Default Values Event Blocks Hat Blocks Looks Blocks List Blocks Monitors More Blocks Motion Blocks Operator Blocks Pen Blocks Reporter Blocks Sensing Sound Blocks Stack Blocks Variables Blocks</p> <p>Project Vocabulary</p> <p>Application and game development Application troubleshooting Boolean logic Conditional programming logic Event handling Interface design Iterative processing</p>	<p>What are blocks and stacks and how do they pertain to programming?</p> <p>What are the basic blocks of Scratch and how do they work together to make a program?</p> <p>How can monitors be used in the program to make it more personal?</p> <p>What are the ten categories of scratch blocks and how are they used in the program?</p> <p>How does Scratch allow a user to get help with code blocks and why is this important?</p>	<p>Bell-ringers for each section</p> <p>Kahoot review game for test</p> <p>Reflection Questions to be written in Journal</p>	<p>Textbook Internet and class website MS Word MS PowerPoint</p>	<p>Mr. Wiggly's Dance-A Quick Scratch Project</p> <p>Chapter Test</p>
--	---	---	---	--	--	---

Computer Programming AC

		Procedural programming Program synchronization Sequential processing Sprite programming Use of variables and lists				
<p>Chapter 5: Moving Things Around</p> <p>“Working with Motion Code Blocks”</p> <p>“Moving and Rotating Sprites”</p> <p>“Setting Sprite Direction”</p> <p>“Repositioning a Sprite”</p> <p>“Changing Sprite Coordinates”</p> <p>“Bouncing Sprites Around the Stage and Controlling Rotation Style”</p> <p>“Keeping Track of Sprite Coordinates and Direction”</p>	<p>NBEA Standards</p> <hr/> <p><u>Programming and Application Development</u></p> <p>Achievement Standard: Design, develop, test, and implement programs and applications.</p> <p>Level 1- 2 Performance Expectations</p> <ul style="list-style-type: none"> Identify and define object-oriented programming terminology Demonstrate the ability to code using object-oriented programming <p>Level 3–4 Performance Expectations</p> <ul style="list-style-type: none"> Identify and explain programming structures Differentiate between source and object code Use scripting languages in application development Apply design principles to programming tasks Develop both procedural and object oriented programs Select and incorporate appropriate compiler Code common tasks using application development tools Test, debug, and document code Maintain and reengineer existing code Develop programs and applications for a variety of platforms Design 3D and gaming environments in relationship to the development of applications <p><u>PA Standards</u></p>	<p>Detecting collisions</p> <p>Motion blocks</p> <p>Predefined rotation axis</p> <p>Scratch cards</p> <p>Sprite Coordinates</p> <p>Sprite direction</p> <p>Sprite rotation</p> <p>X-axis</p> <p>Y-axis</p> <p>Scratch Cards Vocab</p> <p>Animate It</p> <p>Change Color</p> <p>Dance Twist</p> <p>Follow the Mouse</p> <p>Glide</p> <p>Interactive Whirl</p> <p>Keep Score</p> <p>Key Moves</p> <p>Move to a Beat</p> <p>Moving Animation</p> <p>Say Something</p> <p>Surprise Button</p>	<p>What are motion code blocks and how can they be used to move, rotate, and reposition sprites?</p> <p>How does changing sprite coordinates, bouncing sprites around the stage, and controlling rotation style make the application more personal?</p> <p>What are scratch cards and how can they assist a user in learning more about the features of scratch?</p>	<p>Bell-ringers for each section</p> <p>Kahoot review game for test</p> <p>Reflection Questions to be written in Journal</p>	<p>Textbook</p> <p>Internet and class website</p> <p>MS Word</p> <p>MS</p> <p>PowerPoint</p>	<p>Creating the Virtual Scratch Fish Tank</p> <p>Chapter Test</p>

Computer Programming AC

<p>“Taking Advantage of Scratch Cards”</p> <p>1-2 Weeks</p>	<p>15.4.12.H: Use programming languages to develop logical thinking and problem solving skills.</p> <p>15.4.12.I: Compare and contrast programming languages; select most appropriate one to complete a specific task.</p> <p>15.4.12.J: Create a complex computer program to solve a problem.</p>					
<p>Chapter 6: Sensing Sprite Position and Controlling Environmental Settings</p> <p>“Working with Sensing Code Blocks”</p> <p>“Detecting Sprite Collisions and Distance from Objects”</p> <p>“Prompting For and Collecting User Input”</p> <p>“Retrieving Keyboard Input, Mouse Button, and Coordinate Status”</p> <p>“Collecting and Processing Video Input”</p> <p>“Working with a Timer”</p>	<p>NBEA Standards</p> <hr/> <p><u>Programming and Application Development</u></p> <p>Achievement Standard: Design, develop, test, and implement programs and applications.</p> <p>Level 1- 2 Performance Expectations</p> <ul style="list-style-type: none"> ● Identify and define object-oriented programming terminology ● Demonstrate the ability to code using object-oriented programming <p>Level 3–4 Performance Expectations</p> <ul style="list-style-type: none"> ● Identify and explain programming structures ● Differentiate between source and object code ● Use scripting languages in application development ● Apply design principles to programming tasks ● Develop both procedural and object oriented programs ● Select and incorporate appropriate compiler ● Code common tasks using application development tools ● Test, debug, and document code ● Maintain and reengineer existing code ● Develop programs and applications for a variety of platforms ● Design 3D and gaming environments in relationship to the development of applications <p><u>PA Standards</u></p> <p>15.4.12.H: Use programming languages to develop logical thinking and problem solving skills.</p>	<p>Answer code block Code block Collision detection Forever block If block If/Then block If...else code block Sensing blocks Timer True/False value</p>	<p>What is sensing code and how can it detect sprite collisions and distance from objects?</p> <p>Why would prompting user input be important for some applications and how can keyboard input, mouse button, and coordinate status be used?</p> <p>What is a timer and how can it be used in an application?</p> <p>How can the feature of retrieving date, time data, and user’s name be implemented into an application?</p>	<p>Bell-ringers for each section</p> <p>Kahoot review game for test</p> <p>Reflection Questions to be written in Journal</p>	<p>Textbook Internet and class website MS Word MS PowerPoint</p>	<p>Creating Family Scrapbook Application</p> <p>Chapter Test</p>

Computer Programming AC

<p>“Retrieving Stage and Sprite Data”</p> <p>“Retrieving the Date, Time Data, and User’s Name”</p> <p>1-2 Weeks</p>	<p>15.4.12.I: Compare and contrast programming languages; select most appropriate one to complete a specific task.</p> <p>15.4.12.J: Create a complex computer program to solve a problem.</p>					
<p>Chapter 8:Doing a Little Math</p> <p>“Addition, Subtraction, Multiplication, and Division”</p> <p>“Understanding the Mathematical Order of Precedence”</p> <p>“Generating a Random Number”</p> <p>“Comparison Operations”</p> <p>“Performing Logical Comparisons”</p> <p>“Manipulating Strings”</p> <p>“Rounding Numbers and Retrieving Remainders”</p>	<p>NBEA Standards</p> <hr/> <p><u>Programming and Application Development</u></p> <p>Achievement Standard: Design, develop, test, and implement programs and applications.</p> <p>Level 1- 2 Performance Expectations</p> <ul style="list-style-type: none"> Identify and define object-oriented programming terminology Demonstrate the ability to code using object-oriented programming <p>Level 3–4 Performance Expectations</p> <ul style="list-style-type: none"> Identify and explain programming structures Differentiate between source and object code Use scripting languages in application development Apply design principles to programming tasks Develop both procedural and object oriented programs Select and incorporate appropriate compiler Code common tasks using application development tools Test, debug, and document code Maintain and reengineer existing code Develop programs and applications for a variety of platforms Design 3D and gaming environments in relationship to the development of applications <p><u>PA Standards</u></p> <p>15.4.12.H: Use programming languages to develop logical thinking and problem solving skills.</p>	<p>Count Equal to False Greater than Less than Loop Modules Order of precedence Random Remainder Total Total True</p> <p>Mathematical Functions Vocab</p> <p>10^ Abs Acos Asin Atan Ceiling Cos e^ Floor In Log</p>	<p>How can basic math functions be manipulated by Scratch to make performing mathematical equations easier?</p> <p>Why would understanding the mathematical order of precedence in Scratch be important when building a program?</p> <p>What are the built-in mathematical functions, logical comparisons, and manipulating strings in Scratch and why are they important when</p>	<p>Bell-ringers for each section</p> <p>Kahoot review game for test</p> <p>Reflection Questions to be written in Journal</p>	<p>Textbook Internet and class website MS Word MS PowerPoint</p>	<p>Developing the Number Guessing Game Quiz Project</p> <p>Chapter Test</p>

Computer Programming AC

<p>“Working with Built-In Mathematical Functions”</p> <p>1-2 Weeks</p>	<p>15.4.12.I: Compare and contrast programming languages; select most appropriate one to complete a specific task.</p> <p>15.4.12.J: Create a complex computer program to solve a problem.</p>	<p>Sin Sqrt Tan</p>	<p>designing an application?</p>			
<p>Chapter 9: Controlling Script Execution</p> <p>“Introducing Scratch Events and Control Blocks”</p> <p>“Event Programming”</p> <p>“Controlling Script Execution”</p> <p>1-2 Weeks</p>	<p>NBEA Standards</p> <hr/> <p><u>Programming and Application Development</u></p> <p>Achievement Standard: Design, develop, test, and implement programs and applications.</p> <p>Level 1- 2 Performance Expectations</p> <ul style="list-style-type: none"> ● Identify and define object-oriented programming terminology ● Demonstrate the ability to code using object-oriented programming <p>Level 3–4 Performance Expectations</p> <ul style="list-style-type: none"> ● Identify and explain programming structures ● Differentiate between source and object code ● Use scripting languages in application development ● Apply design principles to programming tasks ● Develop both procedural and object oriented programs ● Select and incorporate appropriate compiler ● Code common tasks using application development tools ● Test, debug, and document code ● Maintain and reengineer existing code ● Develop programs and applications for a variety of platforms ● Design 3D and gaming environments in relationship to the development of applications <p><u>PA Standards</u></p> <p>15.4.12.H: Use programming languages to develop logical thinking and problem solving skills.</p>	<p>All Broadcast messages Cloning sprites Conditional logic Control blocks Create loops Endless loop Halt script execution Other scripts in Sprite Pause script execution Program events This script</p>	<p>Why must events and control code blocks be learned to create a script in Scratch?</p> <p>What is event programming and how is it related to initiating script execution and sending and receiving broadcast messages?</p> <p>How are Control blocks linked to pausing script execution, establish loops, or controlling game play?</p>	<p>Bell-ringers for each section</p> <p>Kahoot review game for test</p> <p>Reflection Questions to be written in Journal</p>	<p>Textbook Internet and class website MS Word MS PowerPoint</p>	<p>Developing the Ball Chase Game</p> <p>Chapter Test</p>

Computer Programming AC

	<p>15.4.12.I: Compare and contrast programming languages; select most appropriate one to complete a specific task.</p>					
--	---	--	--	--	--	--

15.4.12.J: Create a complex computer program to solve a problem.