Computer Coding

Southern Cayuga Central School District

Curriculum Map 2022-2023

This document is a work in progress and will be updated throughout the school year!

Title or Topics w/ NYS Standards	Essential Questions & Vocabulary	Content Skills (Activities to cover Essential Questions)	Major Assessments (Tests, Project, etc.)	Time Frame
Introduction to Computational Thinking • **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them. • **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.	• How can I begin to think like a programmer? • What sorts of problem solving skills to programmers use? • Pseudo Code • Algorithm • Top-Down Programming/Design • Careers in Coding	• What makes a good'Vocab' and Concept Focus • programmer? Class Discussion • Blind Draw – Group Activity • What career are you interested in/what is the prevalent language used in this career?	• In-class Assignments	September Intitial Week

Strings
• **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them.
• **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.
• **CCSS.MATH.PRACTICE.MP6** Attend to precision.
	• How can I manipulate text in my program?
	• How can I communicate with a user within my program?
	• What are some basic coding conventions that I should be using when I program?
	• 'Vocab' and Concept Focus
	  ○ for loops, turning, naming guidelines, functions, artistic effects
	  ○ variables, user input, parameters, using *i* in for loops, extended loop control
	  ○ if statements, if/else statements, while loops
		• Examine Code – Group Discussion
		• Projects:
		  ○ Moving efficiently
		  ○ Designing and Communicating Solutions
		  ○ Controlling with vairables
		  ○ making decisions
		  ○ putting together control structures
			• Imbedded Quizes
			• Completion of required pregramming tasks
			• Intro to Python Programming Quiz
				September - October

• **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them.
• **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.
	• How can I use numeric values in my code?
	• How can I break my code down to run differently in different situations?
	• 'Vocab' and Concept Focus
	  ○ Integer output, string output, controlling integer output, controlling string output,
	  ○ print() Syntax:
	  ○ , (comma)
	  ○ *xx (character repetition)
		• Logic Statements – Unplugged Activity
		  ○ Following a chain of logic commands
			• Warm Ups
			• Exit Tickets
			• Project – Simple Guessing Game
			• Project –create using the Makey Makey boards
				October

• **CCSS.MATH.PRACTICE.MP6** Attend to precision. • **CCSS.MATH.PRACTICE.MP7** Look for and make use of structure.	• Boolean Argument • 'Vocab' and Concept Focus ○ Equal, Inequalities, Greater than, Less than, If, Else If, Else, int(), float(), if():, elif():, else:	• Writing if else statements	• Project – Create Guessing game with peer Critique	

• **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them.
• **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically**.**
• **CCSS.MATH.PRACTICE.MP6** Attend to precision.
• **CCSS.MATH.PRACTICE.MP7** Look for and make use of structure.
	• How can I repeat code?
	• 'Vocab' and Concept Focus
	  ○ For loop
	  ○ While Loop
	  ○ Range
	  ○ for i in
	  ○ range():
	  ○ while():
	  ○ range():
		• What would make the "Basic Guessing Game" better? – Group Brainstorm
		• How computers count – an exploration in for loops
		• Update "Basic Guessing Game"
		  ○ For loops vs. While loops
		  ○ Update "Basic Guessing Game" again
			• Project – Improved Guessing Game w/ Peer Critique	October

• **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them.
• **CCSS.MATH.PRACTICE.MP4** Model with mathematics.
• **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.
	• How can I manipulate numeric variables to update my code?
	• Are there ways to get additional built in methods?
	• 'Vocab' and Concept Focus
	  ○ Import
	  ○ Addition (+)
	  ○ Multiplication (*)
	  ○ Subtraction (-)
	  ○ Division (/)
	  ○ Exponents (**)
	  ○ Square Root (math.sqrt())
		• Evaluating Expressions in Python
		• Syntax of mathematical functions
		• Combining what we know – Using mathematical values to make decisions
			• Project –TBD	November

Physical Computing with Raspberry Pi	• What sorts of physical technology can be controlled with code?	• Set Up Raspberry (Crow) Pi Computers • Creating Circuits	• Warm Up • Exit Tickets	November - December

• **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them.
• **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.
• **CCSS.MATH.PRACTICE.MP6** Attend to precision.
	• Raspberry/Crow Pi
	• 'Vocab' and Concept Focus
	  ○ Circuit
	  ○ Button
	  ○ Switch
	  ○ Timer
	  ○ Breadboard
	  ○ LED
	  ○ Sensor
		• Timers
		• Controlling Lights
		• Project – Morse Code
		• The structure of buttons within a circuit
		• Using buttons
		• Project – Reflex Game
			• Project – Morse Code
			• Project – Reflex Game	

Functions / Methods
• **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them.
• **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.
• **CCSS.MATH.PRACTICE.MP6** Attend to precision.
	• How can I create a new method to use throughout my code?
	• 'Vocab' and Concept Focus
	  ○ Function
	  ○ Method
	  ○ Arguments
	  ○ def
	  ○ return
		• Evaluate code – What do you want to be able to change? –Class discussion
		• Writing a function
		• Practice – write a function that prints out a small piece of text art
		• Using arguments in a function
		• Practice – write a function that multiplies a number by 3 and prints it out
		• Using return in a function
		• Practice – write a function that returns True if the argument is an even number and False if it is odd
			• Warm Ups
			• Exit Tickets
			• Project –TBD
				January

Midterm Project • **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them. • **sCCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.	• *Review of all vocabulary and concepts to this point*	• Come up with an idea for a program with a purpose that would make use of the structures you have learned this year. • Write pseudo code describing what your code will actually do in different situations (Project Proposal) • Write the program (Midterm Project) • Peer Code Critique	• Project Proposal • Midterm Project	January

List and More with Strings
 	
• **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them.
• **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.
	• How do I manipulate or check many values at once?
	• 'Vocab' and Concept Focus
	  ○ Index
	  ○ Splicing
	  ➢ [:]
	  ➢ [#:]
	  ➢ [:#]
	  ➢ [#:#]
	  ○ Append
	  ○ Remove
	  ○ In
		• Checking to see if something is in the list
		• Looping through lists
		• Adding and removing items from a list
		• Indexing a list
		• Using splicing with strings
		• Changing one (or more) letters within a string
			• Project - TBD
			• Project - Hangman
				February

Graphing in Python
• **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them.
• **CCSS.MATH.PRACTICE.MP4** Model with mathematics.
• **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically.
	• How can I visualize information using Python?
	• 'Vocab' and Concept Focus
	  ○ Equation
	  ○ Mathplotlib
	  ○ Graph
	  ○ Plt.plot()
	  ○ Linspace()
		• Creating a plot from two lists
		• Creating a plot from an equation and linspace()
		• Adding labels and changing colors on the graph
		• Creating 3D images
		• Determining equations for 3D images & adjusting coefficients
			• Warm Ups
			• Exit Tickets
			• Project – 3TBD
			• Project – TBD
				February - March

Generalizing Code • **CCSS.MATH.PRACTICE.MP1** Make sense of problems and persevere in solving them. • **CCSS.MATH.PRACTICE.MP2** Reason abstractly and quantitatively. • **CCSS.MATH.PRACTICE.MP5** Use appropriate tools strategically. • **CCSS.MATH.PRACTICE.MP7** Look for and make use of structure.	• How can I write code that will work in any possible situation?	• Group Activity – Edit pre-written code so that it is more general	• In-class Assignments	March

• **CCSS.MATH.PRACTICE.M P1** Make sense of • **CCSS.MATH.PRACTICE.M P1** Make sense of problems and persevere in solving them. • **CCSS.MATH.PRACTICE.M P5** Use appropriate tools strategically.	• How can I break down a task into smaller and more manageable parts? • How can I solve a problem with many different possible starting conditions? • Top-down design **Code Quest Practice**	• Top-down design practice • Group Activities – Work on one Code Quest challenge at a time • Group Activities – How many challenges can you complete? (Switch groups every other day) • Group Activity – One Week Challenge (Which team can earn the most points by the end of the week – students make decisions about how to divide up work)	• Project- Previous Code Quest Challenge • Optional: Code Quest Competition? TBD	March-April
AppInventor • **CCSS.MATH.PRACTICE.M P1** Make sense of problems and persevere in solving them. • **CCSS.MATH.PRACTICE.M P5** Use appropriate tools strategically.	• How can I create an app that works on my phone?	• Introduction to App Inventor & block coding • Complete two basic level tutorials • Complete two intermediate level tutorials • Project – Create your own app • Peer Critique	• Project – Create your own app	May
Final Project • **CCSS.MATH.PRACTICE.M P1** Make sense of problems and persevere in solving them. • **CCSS.MATH.PRACTICE.M P5** Use appropriate tools strategically.		•	• • Project-TBD	June