Functions and Parameters

Content Area: **Math**
Course(s): **Introduction to Computer Science**
Time Period: **Marking Period 3**
Length: **4 Weeks**
Status: **Published**

Unit Overview

This unit teaches students how to utilize and write their own fuctions in order to make a program more efficient and readable.

Enduring Understandings

Objectives / Topics Covered

● Functions with and without parameters

● Functions with and without return values

● Nested Control Structures

● Local variables and scope

Solving large and more complex problems using functions

Essential Questions

- Rather than repeat code multiple times, is there a function that can be written?
- When will a function require a parameter and when is one not necessary?

New Jersey Student Learning Standards (No CCS)

8.1.8.CS.1: Recommend improvements to computing devices in order to improve the ways users interact with the devices.

8.1.8.AP.1: Design and illustrate algorithms that solve complex problems using flowcharts and/or pseudocode.

8.1.8.AP.3: Design and iteratively develop programs that combine control structures, including nested loops

and compound conditionals.

8.1.8.AP.4: Decompose problems and sub-problems into parts to facilitate the design, implementation, and review of programs.

8.1.8.AP.6: Refine a solution that meets users' needs by incorporating feedback from team members and users.
8.1.8.AP.7: Design programs, incorporating existing code, media, and libraries, and give attribution.

8.1.8.AP.8: Systematically test and refine programs using a range of test cases and users.

8.1.8.AP.9: Document programs in order to make them easier to follow, test, and debug.

Technology Standards

TECH.8.1.12	Educational Technology: All students will use digital tools to access, manage, evaluate, and synthesize information in order to solve problems individually and collaborate and to create and communicate knowledge.
TECH.8.1.12.A.3	Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.
TECH.8.1.12.A.CS1	Understand and use technology systems.
TECH.8.1.12.A.CS2	Select and use applications effectively and productively.
TECH.8.1.12.B.2	Apply previous content knowledge by creating and piloting a digital learning game or tutorial.
TECH.8.1.12.B.CS1	Apply existing knowledge to generate new ideas, products, or processes.
TECH.8.1.12.B.CS2	Create original works as a means of personal or group expression.
TECH.8.1.12.C.CS4	Contribute to project teams to produce original works or solve problems.
TECH.8.2.12.C.CS1	The attributes of design.
TECH.8.2.12.E.1	Demonstrate an understanding of the problem-solving capacity of computers in our world.
TECH.8.2.12.E.3	Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications, and games).
TECH.8.2.12.E.CS1	Computational thinking and computer programming as tools used in design and engineering.

21st Century Themes/Careers

CAEP.9.2.12.C.3	Identify transferable career skills and design alternate career plans.
CAEP.9.2.12.C.5	Research career opportunities in the United States and abroad that require knowledge of world languages and diverse cultures.

Instructional Strategies & Learning Activities

- 14 functions programming exercises in total

- Using various kinds of functions such as functions with and without parameters, and functions with and without return values

  ○ Example Exercise: Vertical Lines Write a function that draws vertical lines on the graphics canvas. If a line is vertical, then the x-values for the endpoints are the same. The parameters to your function should be the x location, and the length, and all of your lines should start at y position 0.

  ○ Example Exercise: Is it even? Write a function called isEven that returns a boolean of whether or not a value is even or odd. The isEven function should not print anything out or return a number. It should only take in a number and return a boolean. Once you've written this function, write a program that asks the user for integers and prints whether the number they entered is even or odd using your isEven function. You should let the user keep entering numbers until they enter the SENTINEL given.

- 4 challenges using functions to tie everything learned in the Functions and Parameters module together

  ○ Example Exercise: Balloons You should use lines, circles, and random colors to draw a bunch of balloons. All the balloon strings should start two-thirds down the canvas. Each string line should travel upward to a random point and have a circle placed on top of the endpoint. Each balloon should be a random color and have a radius between `MIN_RADIUS` and `MAX_RADIUS`.

Formative Assessments

Quick Friday Quizzes - (2- 3 MC / Short answer questions as well as a programming task).

Summative Assessment

End of Unit Code HS MC Test

Programming Task

Alternate Assessments

Extension exercies and Challenges