FIRIA LABS

LESSON: Functions, parameters, and local variables - part 1

Time: 50 minutes

Overview:

This is part 2 of functions, parameters and local variables. Part 1 gave instructions and practice for mild and medium problems. This lesson gives a quick review of part 1 and then gives spicy practice problems with multiple parameters. It ends with students writing code for a function with multiple parameters.

Objectives:

- I can determine when a variable should be a parameter or a local variable
- I can look at code and identify parameters and local variables
- I can write a function call with multiple argument
- I can use variables of different types as arguments

Standards:

2-CS-03 Systematically identify and fix problems with computing devices and their components.

2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.

3A-AP-17 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.

CSP Framework:

Computational Thinking Practices:

2.B Implement and apply an algorithm.

3.B Use abstraction to manage complexity in a program.

3.C Explain how abstraction manages complexity.

4.C Identify and correct errors in algorithms and programs, including error discovery through testing.

Key Concepts:

- If a function has a parameter, the function call will pass a value to the parameter, called an **argument.**
- **Parameters** and **arguments** need to be in the same order
- An **argument** can be a literal or variable.
- All data types can be variables, and therefore they can be arguments for parameters.

Preparation:

Decide how you will have students work on the activity:

- Work individually or pairs on the assignment
- Work in groups at white boards

Links:

- Instructions slide deck
- Activity for printing
- Assignment (individual)
- Assignment (group work)
- Activity Answers
- Check Your Understanding
- CYU Answers
- Folder with code

Agenda:

- Function review (5 minutes)
- Activity #1 (15 minutes)
- Activity #2 (20-30 minutes)
- Wrap-up & Assessment (5 minutes)

Vocabulary: (review from Mission 9)

- **Function, Parameter, Argument** and **Local Variable**

Assessment:

- Daily reflection journal or Google form
- Assignment completion
- Optional: Check your understanding (combined for part 1 and part 2)
- Code completion

Teaching Guide

✅ This lesson can be taught a couple different ways. Look at the options and decide what is best for your students. The lesson starts with warm-up questions and ends with a reflection. In between are two activities. Option 1 is for students to work in random groups at white boards. Option 2 is for students to work individually or in pairs with the assignment document. Both options will be presented in the teaching guide.

✅ NOTE: This lesson can be done without the assignment document if your students are working in groups at white boards. You can do the warm-up and reflection in a non-print way, or leave it out altogether.

Preparation:

Option 1: Group Work

- Make a copy of **Assignment-Pt 2: Functions, parameters and local variables - board work** for each student in the LMS (if using an assignment document).
- Print the activity problems on paper for the groups to work through (**Activity Prints**).
- The **Check Your Understanding** could be printed and used for individual or pair work – an opportunity for students to try the problems on their own after the group work.
- Be familiar with the answers to the activity problems so you can support your students while they work.

Option 2: Individual or Pair work on assignment

- Make a copy of **Assignment-Pt1: Functions, parameters and local variables** for each student in the LMS.
- Be familiar with the answers to the activity problems so you can support your students while they work.

Warm-up / Function Review (5 minutes)

The warm-up gives students an opportunity to review or see what they remember about functions.

💡 **Teaching tip – warm-up**

- Go through slides #2-3
- These are the same questions as the reflection from Part 1. You can skip the warm-up or change it to meet the needs of your students. You can discuss using a think-pair-share technique.

💡 **Teaching tip – Review**

- Go through slides #4-10 slides with the whole class
- This should be a review, recalling what they did in Part 1
- Spend the time you need to clear up any questions or misconceptions, but don't take too long; try to get to the new material within 5 minutes.

Activity #1 (15 minutes)

💡 **Teaching tip:**

🗝️The key here is to have students identify ALL variables in the function code. Then decide if it is a parameter or a local variable.

💻 How will your students complete the activity?

Option 1:

- Group students randomly and have them work at vertical white boards.
- Use the printouts instead of the slides.

Option 2:

- Students work individually or in pairs
- Use slides #11-14 if needed.
- Students fill out the assignment document

Activity #2 (20-30 minutes)

🗝️Go over slides #15-18 with your students. They discuss multiple parameters, arguments, and using variables as arguments. Then students will complete the final activity.

💻 How will your students complete the activity?

Option 1:

- Group students randomly and have them work at vertical white boards (same or new groups)
- Use a printout
- Students can do all the work at the boards, or you can have them pick partners and do coding at the computer.

Option 2:

- Students work individually or in pairs and go through the assignment document

💡 **Teaching tip - Example for activity 2:**

The coding activity can happen one of two ways:

If you are short on time, students can simply type what the function would look like on the answer document, or write it on the white board.

If you have the time, I suggest the students go to a computer and actually type the code for the function. They can use one of two starter codes, or they can use their own code from Mission 4. Also, they can just start from scratch and do a new program.

If you really have time, they can work on the pseudocode together first, and then go to computers and type the code.

Modify the assignment document to reflect how students will complete activity #2.

Wrap-Up (5 minutes)

The wrap-up is a review of functions, parameters and local variables. If students don't have time for the review, the lesson can be completed without it. If there is time, it is a good reflection for students, but not essential.

As another option, you can assign the **Check for Understanding**. It can be for homework or classwork. It can also be used a later day as a review. Answers are in the folder

✅ You decide what want you want students to turn in for a grade. They may not turn in anything if they participated in group work throughout the class period. Or you can have them do the warm-up and reflection on the assignment document. Or they can do the Check Your Understanding. If students completed the assignment during the class period, they should turn it in. Also, the students may have completed a program with a function that you want them to turn in.

Formative Assessment:

- Daily reflection journal or Google form
- Class discussion on what they learned about abstraction and functions
- Assignment completion
- Check Your Understanding
- Completed program code

SUCCESS CRITERIA:

- ❏ Decide a function name for a section of code
- ❏ Determine parameters for a function
- ❏ Determine local variables for a function
- ❏ Write a function call with multiple parameters
- ❏ Write a function that has multiple parameters