Week of July 21, 2014

Topics for this week: Parameters and Overloading

Activity Checklist

	Read chapter 11 in your course packet.
	Review the slides Parameters and Overloading
	Complete lab #21, due by 11:59pm on Tuesday.
	Complete lab #22, due by 11:59pm on Thursday.
	Complete project #8 and submit it to Canvas before 11:59pm on Sunday. Late programs will lose 20% of the possible points for each day that they are late. If you turn this program in prior to 11:59pm on Saturday, you will receive a 5 point bonus, **if** it meets all of the specifications and gives the correct answers.

Learning Goals

It is expected that you will meet the objectives outlined here by the end of the week. You might want to test yourself to see how well you fare. You can be guaranteed that you will be tested on these concepts on your next exam. By the end of this unit, you should be able to:

- Describe the difference between pass by value and pass by reference.
- Explain when to pass by reference and when to pass by value.
- Correctly write methods that take parameters by value.
- Correctly write methods that take parameters by reference.
- Describe method overloading.
- Correctly write a program that contains overloaded methods.
- Describe the interaction between method overloading and type conversion.
- Correctly construct a driver program to test a method.
- Create stubs in a program to help in development and debugging the program.

Reading Assignment

- The slides on "parameters and Overloading" discuss function overloading and passing parameters by value and by reference.

Key Concepts

You should be sure that you understand the following important ideas in this module:

1. You can write methods so that they pass parameters **by value** or **by reference**.
2. When a parameter is passed by value, a **copy** of the parameter is put on the run-time stack.
3. When a parameter is passed by reference, a reference to the parameter is placed on the stack. A reference is similar to an address - it **refers** to where the original variable is.
4. When parameters are passed by reference, the method may have a side effect. A side effect is when the method changes data defined outside of the method's scope.
5. Multiple methods can be written that have the same name, but take a different number or different types of parameters. Such a method is said to be **overloaded**.


The rule of thumb for passing parameters is to pass primitive data by value. Objects are always passed by reference.

Lab Assignment

This week you should complete labs 21 and 22. These labs will help you to understand the difference between passing a parameter by value and passing a parameter by reference.

Programming Project

This week you should complete your programming project #8. This project will test your ability to create and use methods that pass parameters by value and by reference.