UNIT 03: C++ FUNCTIONS

Lesson 1

- Prototype functions with and without parameters
- Define functions that return a value
- Define functions that make use of reference parameters

DURHAM COLLEGE
SUCCESS MATTERS

CPRG 3202 - INTRO TO C++ - THOM MACDONALD

1

Activity Requirements

- Each of the activities will involve writing programs. You will collaborate in groups of 2 to 3 students, but each of you will write your own programs on your own laptop.
- You will be submitting your source code at the end of the lesson for marks.
- You must be present, in class, to be eligible to participate in the activities, and thus be eligible for these marks.

DURHAM COLLEGE
SUCCESS MATTERS

CPRG 3202 - Intro to C++ - Thom MacDonald

2

Activity 1

- Guided Activity:
  - We will write a simple function that displays a message to the console window in order to understand how to prototype, define, and call a function.
  - Open the starter file *CPRG03-01.cpp*
  - Follow along with the instructor, modifying the code.

CPRG03-01.cpp

CPRG 3202 - Intro to C++ - Thom MacDonald   3

Activity 2A

- Write a simple function that takes a parameter representing a Fahrenheit temperature and returns the equivalent Celsius temperature in order to understand how to pass and return values.
  1. Open the starter file *CPRG03-02A.cpp*
  2. Prototype the *ConvertFahrenheitToCelsius* function based on the documentation provided in the comments.
  3. Define the *ConvertFahrenheitToCelsius* function
  4. Add the function call to main() to complete the program.
  5. Modify the named constant *FAHRENHEIT* and re-run the program a few times to test the function.

CPRG03-02a.cpp

CPRG 3202 - Intro to C++ - Thom MacDonald   4

Activity 2B

- Guided Activity:
  - We will modify our code from Activity 2A in order to understand how to inline functions.
  - Save a copy of your Activity 2A code as *CPRG03-02B.cpp*
  - Follow along with the instructor, modifying the code.

Activity 3

- Write a function called *ConvertTemperature* that takes value parameters representing the temperature and scale to convert from, reference parameters representing the temperature and scale to convert to.
- This function should return true if the scale from was valid (i.e. 'C' or 'F') or false if it was not.
  1. Open the starter file *CPRG03-03.cpp*
  2. Prototype the *ConvertTemperature* function based on the documentation provided in the comments.
  3. Define the *ConvertTemperature* function
  4. Add the function call to main() to complete the program.
  5. Re-run the program a few times with different input to test the function.

CPRG03-03.cpp

CPRG 3202 - Intro to C++ - Thom MacDonald   6

Submission

- Open the *Unit 3- Lesson 1 Activities* Dropbox
- Attach your source code files *individually* (**DO NOT ZIP**) and submit
  - *CPRG01-01.cpp*
  - *CPRG01-02A.cpp*
  - *CPRG01-02B.cpp*
  - *CPRG01-03.cpp*

Summary

- A function is called by giving its name and passing data to it
  - If a variable is an argument in a call, the called function receives a copy of the variable's value
- Common form of a user-written function:

```
returnDataType functionName(parameter list)
{
    declarations and other C++ statements;
    return expression;
}
```

Summary (cont'd.)

- A function's return type is the data type of the value returned by the function
  - If the function does not return a value, it should be declared as a `void` type
- Functions can directly return at most a single data type value to their calling functions
  - This value is the value of the expression in the `return` statement

Summary (cont'd.)

- Reference parameter: passes the address of a variable to a function
  - Reference parameters may be used to modify the argument variable from the calling function
  - Has the effect of *returning* multiple values.

References

Bronson, G. (2012). Chapter 6 Modularity Using Functions. In *A First Book of C++* (4th ed.). Boston, MA: Course Technology.