Lab 9 - CMPS 1043 Computer Science I

Functions

Lesson objectives:
- Demonstrate the use of simple function definition and call
- Demonstrate the correct use of function prototypes
- Distinguish between use of void and int return types

Function: an independent program designed to accomplish a task (generally just ONE task)
- Calculate a value (square root, tangent, sales tax, average, maximum)
- Output (a header, an error message, a return address)

main IS a function, but today we are generally referring to programs OTHER than main.

Any function can access/call any other function (if programmed appropriately). The most common use is probably **main** calling other functions. That will be our emphasis in this lab.

Steps for developing and using functions:
1. Determine functions needed
2. Include prototype for each function [after `#include` statements but before `int main ( )`]
3. Include full function definition after main

Let us write a function called `FindMax` that will accept as parameters 3 integers, N1, `N2`, `N3`, then return the largest (`Big`) of the 3 integers. Since we will return an answer that is an integer, the type of the function must be `int`.

```cpp
// Here is sample format
#include <iostream>
using namespace std;

// Function Prototype; Function name is FindMax;
// Function type is int; Function parameters are 3 integers
int FindMax (int, int, int);     // will determine maximum of 3 integers
                   // can list actual parameters as (int N1, int N2, int N3)
int main ( )
```

The complete function definition is placed after the main function.

```cpp
int main ( )
{
     return 0;
}
int FindMax (int N1, int N2, int N3)
{    int Big;
     if (N1 >= N2 && N1 >= N3)
          Big = N1;
     else if (N2 >= N1 && N2 >= N2)
          Big = N2;
     else Big = N3;
     return Big;
}
```

To **CALL** a function, the command is be placed within another function, in this case, **main**. Since FindMax returns one integer answer, the call must be used in such a way that the answer is saved for future use. For the following call, Biggest will be assigned the value 13.

```
Biggest = FindMax(10, 13, 7);
```

Note that you can use any integer expression as an argument. If x, y & z are integers the following are legal.
```
Biggest = FindMax(x, y, z);
Biggest = FindMax(10, x+7, z*7);
Biggest = FindMax(10-x, abx(z), x+y*(x+z));   //abs is absolute value
```

Class Exercise:
Part 1: In C++ type in the code given above into a new project. Complete main so that it will call FindMax using the command given then print Biggest. Correct all errors until it correctly prints 13.

Part 2: Modify main declaring 3 integers, A, B, and C. Ask the user to input 3 integers for A, B, and C. Then call FindMax to determine and print the biggest of the 3 integers.

```
Biggest = FindMax(A, B, C);
```

Part 3: Develop another function called PrintVal that accepts as a parameter one integer X. It then prints the following statement and the value of X. (E.G. if X = 100)

```
The answer is 100
```

Include the prototype for PrintVal immediately below the prototype for FindMax.
Include the complete function definition for PrintVal immediately below the definition of FindMax.
Modify main so that it calls PrintVal to print the largest value AFTER it receive the answer from FindMax.
Execute and correct until the Project runs correctly.

Now, Modify FindMax so it calls PrintVal to print the largest value BEFORE returning the answer to main.
Execute again. What is wrong??
Change the order of the prototypes then execute until the priject reuns correctly.

DOCUMENTATION of Functions:
Every function must have a set of comments describing the function as follows:
// Input: state function parameters and what they represent
// Processing: state task that function will accomplish
// Output: return value returned to calling function
For function PrintVal, the dcoument should look similart to the following:

```
// Function PrintVal
// Input:  accepts one integer parameter
// Process: print input parameter to the screen
// Output: none
// Additional comments as necessary
void PrintVal (int X)
```

The documentation can be placed:
- Above the function prototype
- Above the function definition ** Dr. Halverson's students use this location

Class Exercise (continued)
Add COMPLETE documentation to your project then print off to turn in to your lecture instructor. (code and output)