Lab # 13: Functions - III

EC-102 – Computer Systems and Programming

Usman Ayub Sheikh

School of Mechanical and Manufacturing Engineering (SMME),
National University of Sciences and Technology (NUST)

Friday 20th November, 2015

Outline

Reference Arguments

- An alias, a different name for a variable
- One of the most important uses is in passing arguments to functions

Passing by Value vs Passing by Reference

- When arguments are passed by value, the called function creates a new variable of the same type as the argument and assigns the argument's value to it
- The function cannot access the original variable in the calling program, only the copy it created
- It is useful when the function does not need to modify the original variable in the calling program
- It offers insurance that the function cannot harm the original value

Passing by Value vs Passing by Reference

- Passing arguments by reference uses a different mechanism
- Instead of a value being passed to a function, a reference to the original variable, in the calling program, is passed
- An important advantage is that the function can access the actual variables in the calling program
- Provides a mechanism for passing more than one value from the function back to the calling program

Passing Simple Data Types by Reference

```
1  // ref.cpp
2  // demonstrates passing by reference
3  #include <iostream>
4  using namespace std;
5  void intfrac(float, int&, float&); //declaration
6
7  int main()
8  {
9      float number, fracpart;
10     int intpart;
11
12     do {
13         cout << "\nEnter a real number: "; //number from user
14         cin >> number;
15         intfrac(number, intpart, fracpart); //find int and frac
16         cout << "Integer part is " << intpart //print them
17         << ", fraction part is " << fracpart << endl;
18     } while( number != 0.0 ); //exit loop on 0.0
19     return 0;
20 }
```

Passing Simple Data Types by Reference

```
21 // intfrac()
22 // finds integer and fractional parts of real number
23 void intfrac(float n, int& intp, float& fracp)
24 {
25     intp = n;
26     fracp = n - intp; //subtract integer part
27 }
```

Exercise

Write a function that

- takes a reference to a binary number (base 2) as an argument, and
- converts that number to a decimal number (base 10).

Write a program that

- exercises this function by obtaining a binary number from the user,
- and printing out the equivalent decimal number.