Discussion #5

Ch.5 – Part 1

Nov. 2013
Function:

double cube_volume(double side_length)
{
    double volume = side_length * side_length * side_length;
    return volume;
}

Call

int main()
{
    double result1 = cube_volume(2);
    double result2 = cube_volume(10);
    cout << "A cube with side length 2 has volume "
         << result1 << endl;
    cout << "A cube with side length 10 has volume "
         << result2 << endl;
    return 0; }

Functions Without Return Values – The void Type

void box_string(string str)

void functions are used only to do a sequence of instructions.

void box_string(string str)
{
    int n = str.length();
    int n = str.length();
    if (n == 0)
    {
        return;
    }
    . . .  // None of these statements
    // will be executed

Review Exercises

R5.5 Consider these functions:

double f(double x) { return g(x) + sqrt(h(x)); }

double g(double x) { return 4 * h(x); }

double h(double x) { return x * x + k(x) - 1; }

double k(double x) { return 2 * (x + 1); }

Without actually compiling and running a program, determine the results of the
following function calls:

c. double x3 = k(g(2) + h(2));
Sol.

\[ x_3 = k(g(2) + h(2)) \]
\[ = k(36 + 9) \quad \text{\{from above examples\}} \]
\[ = k(45) \]
\[ = 2 \times (46) \]
\[ = 92 \]

\{K(2)= 2*3=6\}

H(2)= 4+5=9

G(2)=4*9=36

X3 = k(36+9) = k(45)=46*2 = 92\}

R5.8 For each of the variables in the following program, indicate the scope. Then determine what the program prints, without actually running the program.

```cpp
int a = 0; // a is a global variable
int b = 0; // b is a global variable
int f(int c) // this parameter variable c is local to function f
{
    int n = 0; // this n is local to function f
    a = c;
    if (n < c)
    {
        n = a + b;
    }
    return n;
} // n and c no longer defined

int g(int c) // This parameter variable c is local to g
{
    int n = 0; // This n is local to function g
    int a = c; // This a is local to function g (it “shadows” the
    // global variable a)
    if (n < f(c))
    {
        n = a + b;
    }
```
R5.16 Consider the following function that is intended to swap the values of two integers:

```cpp
void false_swap1(int& a, int& b) {
    a = b;
    b = a;
}
int main() {
    int x = 3;
    int y = 4;
    false_swap1(x, y);
    cout << x << ' ' << y << endl;
    return 0;
}
```

Why doesn’t the function swap the contents of x and y? How can you rewrite the function to work correctly?

**Sol.**

The problem in the false_swap1 function is that the original value of a is lost before you can transfer it to b. A temporary variable is required or another way to perform the swap must be performed. The correct version of the function would be:

```cpp
void swap1(int& a, int& b) {
    int temp = a;
    a = b;
    b = temp;
}
```
P5.3 Write the following functions and provide a program to test them.

a. double smallest(double x, double y, double z), returning the smallest of the arguments

Sol.

```cpp
#include <iostream>
using namespace std;

double smallest(double x, double y, double z)
{
    if (x <= y && x <= z)
        return x;
    else if (y <= x && y <= z)
        return y;
    else
        return z;
}

int main()
{
    cout << "Enter three numbers: ";
    double a;
    double b;
    double c;
    cin >> a >> b >> c;

    cout << "The smallest number is: " << smallest(a, b, c);

    return 0;
}
```

See You At The Next Discussion