Prelab #7 – CSC 150

Purpose:

1. Interpret function prototypes to determine correct function usage
2. Write valued and void functions
3. Determine the scope of variables

I. Given the following function prototypes, indicate if the function calls in the statements are valid (V) or invalid (I). Consider each call independently (this is not a complete working program! Some calls may give compiler warnings – consider these as Valid).

```
//function prototypes
void coolfunc( int x, int y );
int sum( int x, int y );
float func3( float z );
void changer( float &p );

//variable declarations
int i = 0;
float f = 1.0;

//function calls
```

a. V i = coolfunc(10 , 12);
b. V coolfunc(10 , i);
c. V sum(10 , 12);
d. V cout << sum(i , 2);
e. V i = sum(3, coolfunc(4 , 5));
f. V changer (f);
g. V i = sum(10 , 12) / func3(4.0);
h. I cout << "Hello world. " << func3(4.0 , 3.0) << endl;
i. I changer (37.6);
j. V f = func3(sum(2 , 4));
k. I i = summ(4 , 2);
l. I coolfunc(32);
m. V coolfunc (i, sum(i, 5));
n. V changer (i);
o. V i = changer(f);

II. Write definitions of the following functions, implementing the actions described for them.

double mid (double a, double b, double c) //Returns the value that is the middle of
//a, b and c. That is, the value that is not the
//largest and not the smallest

```
{
  int x
  x=(a+b+c)/2
  return(x)
}
```

void get_3_rand (int &num1, int &num2, int &num3, int limit) //get 3 random numbers,
//ranging from 1 up to (including) the limit value. Assume srand( ) has already been
//called

III. Read through the program PL7_scoping.cpp available in the Prelab folder on the course website. Determine the value that will be printed at each output as the program runs. Write your answers in the blanks below. Then put the file into a project and verify your answers.

local x in outer scope of main is: 5
local x in inner scope of main is: 7
local x in outer scope of main is: 5
local x is: 25 after entering a.
local x is: 26 before exiting a.
local static x is: 50 on entering b.
local static x is: 51 on exiting b.
global x is: 1 on entering c.
global x is: 10 on exiting c.
local x is: 25 after entering a.
local x is: 26 before exiting a.
local static x is: 51 on entering b.
local static x is: 52 on exiting b.
global x is: 10 on entering c.
global x is: 100 on exiting c.
local x in main is: 5
x in the main program before calling d is : 5
The parameter x on entering d is : 5
The parameter x on exiting d is : 6
x in the main program after calling d is : 5
x in the main program before calling e is : 5
The parameter x on entering e is : 5
The parameter x on exiting e is : 6
x in the main program after calling e is : 6
x in the main program before calling f is : 6
global x on entering f is : 100
global x on exiting f is : 101
x in the main program after calling f is : 101
x in the main program before calling g is : 101
parameter x on entering g is : 101
parameter x on exiting g is : 102
x in the main program after calling g is : 102