CISC1600/1610
Fall 2022
Self-practice Questions on functions

1.Which of the following statements about variables is true?

(A) The same variable name can be used in two different functions.
(B) The same variable name cannot be used for two different variables in a single function.
(C) You should use global variables whenever possible.
(D) A variable is visible from the point at which it is defined until the end of the program.

2. Which of the following is correct about a global variable?

(A) It is declared before all the functions in a program.
(B) It is visible to all the functions declared after it.
(C) It is declared in the main function.
(D) It is declared within the scope of a function.

3. What is the syntax error in the following function definition?

```
string parameter(double r)
{
   double result;
   result = 2 * 3.14 * r;
   return result;
}
```

(A) The function does not return the value result.
(B) The function does not specify result return type.
(C) The variable result is set but never used.
(D) The value that is returned does not match the specified return type.

4. For each of the variables in the following program, indicate the variable's scope by stating the lines in the code where it's visible. Then determine what the program displays in the terminal.

```
1.  int a=0;
2.  int b=0;
3.  int f (int c)
4.  {
5.     int n = 0;
6.     a = c;
7.     if (n < c)
8.        n = a + b;
9.     return n;
```

```
10. }
11.
12. int g (int c)
13. {
14.     int n = 0;
15.     int a = c;
16.     if (n < f(c))
17.         n = a + b;
18.     return n;
19. }
20.
21. int main ()
22. {
23.     int i = 1;
24.     int b = g(i);
25.     cout << a + b + i << endl;
26.     return 0;
27. }
```

5. Consider the following code segment that is intended to swap the values of two integers. What's the output of the program?

```
1.  void swap ( int a , int b)
2.  {
3.     int temp = a;
4.     a = b;
5.     b = temp;
6.  }
7.
8.  int main()
9.  {
10.    int x = 3;
11.    int  y =4;
12.    swap (x, y);
13.    cout << x <<" " << y << "\n";
14.    return 0;
15. }
```

6. **Match the following terms with the definitions given below.**
    a. Argument
    b. Parameter
    c. Function call
    d. Function prototype

e. Function definition

f. Local variable

g. Value parameter

h. Reference parameter

1) A function declaration without a body
2) A parameter that receives a copy of the argument's value
3) A variable declared in a function heading
4) A function declaration with a body
5) A value or expression listed in a call to a function
6) A statement that transfers control to a function
7) A parameter that receives the address of the argument
8) A variable declared within a block

7. When calling a function that takes no parameter, such as the following, you do not need to put ( ) after the function name. True or false?

```
int fun1( );

int main( )
{
    func1;
}

int func1 ( )
{
      int a;
      cin >> a;
      return a;
}
```

8. When can you use a function call in an expression? For example, under what situation the following is legal: double value = funca( );

9. Find out syntax errors in the following code:

```
void GetTwoValue (int & x, int & y);

int main ( )
{
       int a, b;
       GetTwoValues (3, b);
       GetTwoValues (a+b, b);
}

void GetTwoValues (int & x, int & y)
{
      int tmp1, tmp2;
      cin >> tmp1 >> tmp2;
      x= tmp1;
      y = tmp2;
}
```

10. Find out logic or syntax errors in the following code:

```
bool CheckDate (int year, int month, int day);
//return true if the date given by parameter month/day/year is a valid date;
//return false otherwise

int main ( )
{
       int mon, day, yr;
       cout <<"Enter your birthday (MM DD YYYY):";
       cin >> yr > mon >> day;

       CheckDate (mon, day, yr);
       If (CheckDate)
          cout <<"All right!\n";
       else
         cout <<"Wrong date!\n";
}
```

11. Trace the execution of the following programs/code segment to find out the output of the function:

a) //using swap

```
void swap (int & x, int & y);

int main ()
{
      int value1, value2;
      value1 = 2;
      value2 = 10;
      cout <<"value1=" << value1 << endl;
            << "value2=" << value2 <<endl;
      swap (value1, value2);
      cout <<"After function call: " ;
      cout <<"value1=" << value1 << endl;
            << "value2=" << value2 <<endl;

}

void swap (int & x, int & y)
{
     int tmp = x;
     x = y;
     y = tmp;
}
```

b) //calculate sales tax

```
void AddTax (double & amt_due, double rate);

int main ( )
{
          double total_order = 120.00;
          cout <<"Total order: " << total_order <<endl;
          AddTax (total_order, 0.05);
          cout <<"With tax (5%), total due:"  << total_order<<endl;
}

void AddTax (double & amt_due, double rate)
{
    amt_due = amt_due + amt_due*rate;
}
```

Now try to trace the program, assuming the first parameter amt_due to AddTax function is passed by value, i.e., we remove the & between "double" and "amt_due"?

13. Practice writing functions… Noting:

a) function name should be descriptive (reflect what the function does)

b) Pass-by-value parameters are used to accept input.

c) If the function returns/generates one result, use return type/statement to return it

d) If the function generates/produces multiple results, use pass-by-reference parameter to make them available to the caller

e) If the function needs to modify parameter's value, use pass-by-reference

a. Write a function that calculates and returns the maximum value among three integer numbers.

b. Write a function that calculates and "returns" both the maximum and minimum values among three integer numbers.

c. Write a function that displays a square consists of * with a given side length, for example, if called with side length 4, the following will be displayed:

```
****
****
****
****
```

d. Write a function that displays a hollow square, for example, if called with side length5, it displays:

```
*****
*   *
*   *
*   *
*****
```

14. For each of the variables in the following program, indicate the variable's scope by stating the lines in the code where it's visible. Then determine what the program displays in the terminal, assuming the user enters -2 at the prompt message "Enter an integer:".

```
1.   int a = 0;
2.    int b = 0;
3.
4.    int sum (int c)
5.    {
6.       int sum=0;
7.       for (int a=0; a<c;a++)
8.            sum += a;
9.       return sum;
10.   }

11.   int absolute (int c)
12.   {
13.       if (c > 0)
14.            return c;
15.      else
16.           return –1*c;
17.   }

18.   int main ()
19.   {
20.       cout << "Enter an integer: " ;
21.       cin >> a;
22.       cout << sum (absolute (a)) << endl;
23.       return 0;
24.   }
```

15. Short programming assignment (submit this using submit1600, name your file triangle.cpp): write a function that display a triangle of a certain side length, using a given character. For example, one can call the function to display a triangle made up of # with side length 3

```
  #
 ###
#####
```

Or a triangle made up of + of side length 4:

```
   +
  +++
 +++++
+++++++
```

Hint: Try to think through the problem, possibly by writing out a pseudocode, i.e., English description of the procedure. For example, to print out a triangle of side length given by *side_length*, using character *ch*:

Initialize the number of leading spaces in first line (Hint: you can try to figure this out, it is related to *side_length*)
Initialize the number of character ch to display in first line
Repeat the following for side_length time:

a. display a certain number of space character (hint: this can be done using a for loop)
b. display a certain number of character ch (hint: this can be another for loop)
c. display a newline character
d. update the number of space characters, and ch for next line (Hint: study the shape of triangle to find out how)

16.. Trace the execution of the following programs/code segment to find out the output of the program: Note please show your work.

```
void AddTax (double & amt_due, double rate);
int main ( )
{
        double total_order = 120.00;
        double rate = 0.05;
        cout <<"Total order: " << total_order <<endl;
        AddTax (total_order, rate);
        cout <<"With tax  total due:"  << total_order<<endl;
        cout <<"Tax rate is :" << rate << endl;
}

void AddTax (double & amt_due, double rate)
{
    rate = rate+1.0;
    amt_due = amt_due*rate;
}
```

17. Given the following function:

```
int InputItemNumber (string itemname)
{
```

```
  int input;
  do
   {
     cout << "Enter the number of " << itemname << ": ";
     cin >> input;
   }
  while (input < 0);

  return input;

}
```

When called, the function will loop until a non-negative value is entered.
Now please modify this function so that user can choose to give up, when a negative number is entered.

The function needs to pass/produce two values to the caller:

- A Boolean value that indicates whether a valid number has been entered or not
- The number entered by the user

18. What's the ouput of the following code?

```
int pow(int base, int power)
{
   int result = 1;
   for (int i = 0; i < power; i++)
   {
      result = result * base;
   }
   return result;
}

int main()
{
    cout << pow(pow(2, 2), 2) << endl;
   return 0;
}
```