Tutorial Letter 104/0/2022

Introduction to Programming I

COS1511

Year:2022

School of Computing

This tutorial letter contains important information about your module.

BARCODE

Assignment 2: 2022 SUBMISSION:

Electronically via *myUnisa*

Please note that we automatically give five days extension for this assignment. It will be to your own advantage to check after a few days whether the assignment has been registered on the system. If you have not completed the assignment by the extension date, submit whatever you have completed – you will get marks for everything that you have done.

DUE DATE	**16 May 2022**
UNIQUE NUMBER	**830199**
EXTENSION	**There is an automatic extension until 10 April. You do not need to phone or send an e-mail to request automatic extension**
TUTORIAL MATTER	**Study Guide, Lessons 17 – 23**
CONTRIBUTION WEIGHT TO YEAR MARK	**20%**
QUESTIONS	**Practical exercises**

If *myUnisa* is off-line when you want to submit the assignment, you need not contact us, because we will be aware of it. Simply submit it as soon as *myUnisa* is available again.

Question 1

Write function headers for the functions described below:

(i) The function `check` has two parameters. The first parameter should be an integer number and the second parameter a floating point number. The function returns no value.

(ii) The function `mult` has two floating point numbers as parameters and returns the result of multiplying them.

(iii) The function `time` inputs seconds, minutes and hours and returns them as parameters to its calling function.

(iv) The function `countChar` returns the number of occurrences of a character in a string, both provided as parameters.

Question 2

Find the error(s) in each of the following program segments and explain how the error(s) can be corrected:

(i)
```
int function1()
{

    cout << "Inside function function1 " <<
```

```
            endl; int function2()
            {

                cout << "Inside function function1 " << endl;
            }
        }
```

(ii)
```
        int sum(int x, int y)
        {

            int result;
            result = x +
            y;
        }
```

(iii)
```
        int computeProd(int n)
        {

            if (n == 0)
                return 0;

    else

}

n * computeProd(n - 1);
```

(iv)
```
        void aFunction(float a)
        {

            float a;
            cout << a << endl;
        }
```

(v)
```
        void theProduct()
        {

            int
            a;
            int
            b;
            int
            c;
            int result;

            cout << "Enter three integers " <<
            endl; cin  >> a >> b >> c;
            result = a * b * c;
            cout << "Result is " << result <<
            endl; return result;
        }
```

(vi)
```
float calculateSquare(float number)
{

    return number * number;
}
```

Question 3

Write the functions as described below. You need not submit programs that use these functions.

(i) Write a function that returns the cube of the integer passed to it. For example `cube(2)` will return `8` and `cube(3)` will return `27`.

(ii) Write a void function that receives four `int` parameters: the first two by value and the last two by reference. Name the formal parameters `n1`, `n2`, `sum` and `diff`. The function should calculate the sum of the two parameters passed by value and then store the result in the first variable passed by reference. It should calculate the difference between the two parameters passed by value and then store the result in the second parameter passed by reference. When calculating the difference, subtract the larger value from the smaller value. Name the function `calcSumAndDiff`.

(iii) Write a function `void rectangle(int w, int h)` to print an open rectangle of asterisks (`*`). The parameters `w` and `h` are the width and the height of the rectangle, expressed in number of asterisks.

(iv) Write a function, `computePrice` that receives two parameters: the first one is a character variable indicating the size of the pizza (S, M or L) and the second an integer variable indicating the number of toppings on the pizza. It then computes the cost of the pizza and returns the cost as a floating point number according to the rules:

- Small pizza = R50 + R5.50 per topping
- Medium pizza = R70 + R6.50 per topping
- Large pizza = R90 + R7.50 per topping

Question 4

The post office needs a program that reads in postal address data and then displays the data in a neat format.

- Declare four string variables `name`, `addr1`, `addr2` and `postalCode`.
- First, the `main` function of the program must input the data by means of a function called `inputData`.
- Second, the `main` function calls the function `displayData` to display the name and address as follows:

```
Mr R.S. Bopape
P.O. Box
50741
Sandton
2146
```

Submit the program and the output.

Question 5

Write a program that calculates the average of a group of test scores, where the lowest score in the group is dropped. For example: if the user enters the values 65, 43, 78, 67 and 64, the output will be:

```
After dropping the lowest test score, the test average
                        is 68.50
```

This average is calculated by dropping the lowest score which is 43 and dividing the sum of the remaining four values 65, 78, 67 and 64, which is 274, by 4.

It should use the following functions:

- An `int` function `getScore` with no parameters, to ask the user for a single test score, validate it and return the score. The function should be called by the `main` function once for each of the five scores to be entered. Do not accept test scores lower than 0 or higher than 100.
- An `int` function `findLowest` that is passed the five test scores and then finds and returns the lowest of the five scores passed to it. It is called by function `calcAverage`, to determine which of the five scores to drop.
- A `float` function `calcAverage` that is passed the five test scores and then calculates and returns the average of the four highest scores.
- A `void` function `displayOutput` that is passed the average score; it then displays the average of the test scores. Display two digits after the decimal point.

You must submit the four functions you have developed as well as output using the following 5 sets of input data:

65 24 80 73 51
66 38 84 69 59
72 52 81 23 53
65 28 72 63 65
65 55 75 68 62

Submit the program and the output.

Question 3

Write a program that will compute the volume of a room. User inputs are the height, width and length of a room. For example, if the user inputs the height as 3, the width as 4 and the length as 5 the volume will be 3 * 4 * 5 = 60. You have to declare three functions: one for input; one to do the calculation; and one for output.

- The input function `getData` has to input the height, width and length into the variables `theHeight`, `theWidth` and `theLength`.
- The calculation function, `calculateVolume` will receive three parameters that represent the height, width and length, do the calculation and return the volume of the room.
- The output function `displayOutput` has to display the height, width and length entered by the user as well as the volume.
- The program also has to indicate the size of the room as small, medium or large. If the volume is less than 100, the size is small, between 100 and 500 the size is medium and greater than 500 the size is large. For example:

```
The volume of a room with height 3, width 4 and length
5 is 60. Size: Small
The volume of a room with height 10, width 14 and
length 12 is 1680.
Size: Large
The volume of a room with height 7, width 7 and length
7 is 343. Size: Medium
```

- The `main` function should include a `for` loop that allows the user to repeat the calculation of the volume for new input values five times.

You must submit the three functions and the `main` function you have developed, as well as output produced by the following input data:

```
3 4 5
10 14 12
8 7 7
12 14 8
15 8 12
```

Question 7

Draw a series of variable diagrams for the program below. Use the conventions of the Study Guide.

```
1    // variable diagrams revisited
2    #include <iostream>
3    using namespace std;
4    const int C = 200;
5    int func1(int n, int n1)
6  {
7         n += 3;
8         n1 -= n;
9         return 2 + n + n1
* C; 10  }
11  void func2(int n, int & n1)
12  {
13        n = C * n1;
14        n1 = n -
n1;
15  }
16  void func3(int & n, int &
n1) 17  {
18        int k;
19        k = n1 + 3;
20        n = k * 30;
21        n1 = n + 2 * k;
22  }
23  int main(
) 24  {
25        int n, m, j;
26        n = 5;
27        m = 10;
28        j = func1(n, m);
29        n = 15;
30        m = 20;
31        func2(n, m);
32        n = 25;
33        m = 30;
34        func3(n,
m);
35
36        return
0; 37  }
```