Tutorial Letter 102/1/2018

Introduction to Programming I
COS1511

Semester 1

School of Computing

This tutorial letter contains Assignment 2 for Semester 1.

BARCODE

Define tomorrow.

UNISA | university of south africa

CONTENTS

1   INTRODUCTION

Dear student,

Hope you are well. This tutorial letter contains Assignment 2. Assignment 2 contributes 40% towards your semester mark and requires you to write and submit programming code.

2   ASSIGNMENT 2

Assignment 2 is a practical assignment and thus must be written and submitted.

- We urge you to do and submit this assignment; otherwise you will find it very difficult in the examination.
- For this assignment you have to write, compile and run programs using the software that you have installed.
- Students must submit this assignment via *myUnisa*.
- Assignment 2 must be submitted in **PDF format**. See Additional Resources on the COS1511 course website for instructions on how to create an assignment as a PDF file.
- Keep to the submission date for the assignment as stated in the study plan of this letter.
- We do not necessarily mark all questions. You will get 0% if you do not submit the questions that are marked.
- The semester system does not allow for late submission of assignments. However, we give 4 days automatic extension for Assignment 2. This means that you may submit the assignment up to 4 days after the due date, without making special arrangements with the lecturers. Although *myUnisa* will give you a message to say that your assignment is late, the assignment will still be accepted by the *myUnisa* system.
- The programs must be written in C++. You may not use any other high-level language for COS1511. Your programs must follow the programming style used in the Study Guide. In the assignments and the examination, marks are awarded for **programming comments, programming style, syntax and logic.**
- The input data as specified in the questions will be used to test your programs.
- No marks are allocated for programs that do not compile.

3   CALCULATION OF THE SEMESTER MARK

The marks that you obtain for Assignments 1, 2 and 3 form the semester mark for COS1511. The semester mark forms 20% of the final mark for the module. The weights of the COS1511 assignments are indicated in the table below:

Assignment number	Weight
1	30%
2	40%
3	30%

An example follows: Suppose a student gets 60% for Assignment 1, 45% for Assignment 2 and 65 for Assignment 3. In order to calculate the semester mark, the mark obtained for the specific assignment is multiplied by the weight. This then forms part of the 20% that the semester mark contributes to the final mark. Therefore:

Assignment	Marks obtained	Weight	Contribution to semester mark	
1	60%	30%	60/100 x 30/100 x 20	3.6
2	68%	40%	68/100 x 40/100 x 20	5.4
3	65%	30%	65/100 x 30/100 x 20	3.0
TOTAL				12.0

In this example the student has a semester mark of 12.0 out of 20. **The semester mark will not form part of the final mark of a supplementary examination.**

4   DUE DATES OF ASSIGNMENT

The table below gives the due dates of the assignments for this module.

Assignment	Due Date 1st semester	Weight
~~1~~	~~19 March~~	~~30%~~
2	9 April	40%
3	26 April	30%

5   SUBMISSION OF ASSIGNMENT 2

Submit assignment 2 (as a .pdf file) via *myUnisa.* No assignments in the wrong format can be accepted.

For detailed information and requirements as far as assignments are concerned, see *Studies@Unisa*, which you received with your study package. Follow the instructions given in Tutorial Letter COSALLF/301/4/2018, as well as the brochure *Studies@Unisa*, when submitting your assignments. The URL for *myUnisa* is: http://my.unisa.ac.za/. Instructions on how to register to become a *myUnisa* user, and how you should format your assignments before you submit them electronically, are given on the website. The two most important things to remember are that your submission must consist of a single text file, and that you may submit an assignment only once.

To submit an assignment through *myUnisa*:

- go to *myUnisa*
- log in with your student number and password
- select the module
- click on assignments in the menu on the left-hand side of the screen
- click on the assignment number that you wish to submit
- follow the instructions

PLEASE NOTE: Assignments can be tracked (e.g. whether or not the University has received your assignment or the date on which an assignment was returned to you) on *myUnisa*.

6   ASSIGNMENT 2: 1ST SEMESTER

SUBMISSION: **Electronically via *myUnisa***

Please note that we automatically give four 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.

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.

DUE DATE	**9 April**
UNIQUE NUMBER	745328
EXTENSION	There is an automatic extension until **13 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 SEMESTER MARK	40%
QUESTIONS	Practical exercises

QUESTION 1:

Question 1a

Suppose we want to input and validate the age of students that qualify for an internship, as well as the final mark obtained for the examination, in a `while` loop. To qualify, the student should be younger than 30 with a final mark of more than 65%. Read in values util a suitable candidate is found. Display appropriate messages, wheter successful or not. The variable names are `age` and `finalMark` respectively. Complete the `while` loop below. You only have to write down the completed `while` loop.

```
cout << "Enter age: ";
cin  >> age;
cout << "Enter final mark for exam: ";
cin  >> finalMark;

while (…………………) //complete the condition
{

  //complete the while loop
}
```

submit   You only have to submit the completed `while` loop.

Question 1b

State what output, if any, results form each of the following statement by first working it out on paper and then including it is a program. Submit a completed table as below:

	CODE	OUTPUT

example	`for (int i = 0; i < 10; i++)`
	`    cout << i;`
	`cout << endl;`
		`0123456789`

a.	`for (int i = 1; i <= 1; i++)`
	`    cout << "*";`
	`cout << endl;`	

b.	`for (int i = 2; i >= 2; i++)`
	`    cout << "*";`
	`cout << endl;`	

c.	`for (int i = 1; i <= 1; i--)` `    cout << "*";`	

	`cout << endl;`	

d.	`for (int = 12; i >= 9; i--)`
	     `cout << "*";`
	 `cout << endl;`	

e.	`for (int i = 0; i <= 5; i++)`
	     `cout << "*";`
	 `cout << endl;`	

f.	`for (int i = 1; i <= 5; i++)`
	     `cout << "*";`
	     `i = i + 1;`
	 `cout << endl;`	

submit   Redraw table and submit completed table.

Question 1c

Include the `for` loop below in a small program and complete the program. The loop should be executed 10 times. Do not change the `for` loop below. Compile and run your program to see for yourself that it works. You do not have to submit this program and output.

```
for (int i = 0; i <= n; i++)
    if (i < 5 && i != 2)
        cout << 'X';
```

Now convert the `for` loop into a `while` loop and add any variable initialisations that you think are necessary. Compile and run your program.

submit   Submit only the program containing the while loop and its output.

Question 1d

When running the program given, it is supposed to give the output bewlow. However, the program contains erros that prevent it from compiling and/or running. Correct the program so that it works properly.

The output:

```
Please enter 10 integers, positive, negative, or zeros.
The numbers you entered are:
2
7
-4
-3
0
7
4
0
-9
-4

There are 6 evens, which includes 2 zeros.
The number of odd numbers is: 4
```

The program:

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

const int LIMIT = 10;

int main ()
{
    float counter;
    int number;

    int zeros;
    int odds;
    int evens;

    cout << "Please enter " << Limit << " integers, "
         << "positive, negative, or zeros." << endl;

    cout << "The numbers you entered are:" << endl;

    for (counter = 1; counter <= LIMIT; counter++)
    {
        cin << number;

        switch (number / 2)
        {
            case 0:
                evens++;
                    if (number = 0)
                    zeros++;
```

```
            case 1:
                case -1:
                odds++;
        }
    }

    cout << endl;

    cout << "There are " << evens << " evens, "
         << "which includes " << zeros << " zeros."
         << endl;
    cout << "The number of odd numbers is: " << odds
         << endl;

    return 0;
}
```

submit   Submit only the program, not the output.

QUESTION 2:

In this question, we describe the problem and then you have to decide yourself how you are going to tackle it.

Question 2a

The cost of renting a romm at a hotel is R900 per night. For special occasions, such as a wedding or conference, the hotel offers a special discount as follows:

- if the number of rooms booked is at least 10, the discount is 10%;
- if the number of rooms booked is at least 20, the discount is 20%;
- if the number of rooms booked is greater or equal 30, the discount is 30%;

In addition, if rooms are booked for at leas three days, there is an additional 5% discount.

Write a program that promps the user ro enter the cost of renting one room, the number of rooms booked, the number of days the rooms are booked and theslaws tax (as a percent).

Display the output as follows:

```
        Please enter the following:
                cost per room: 1000
            sales tax per room: 10
          the number of rooms: 35
                number of days: 2

        The total cost for one room is R1000
        The discount per room is 30%
        The number of rooms booked: 35
        The total cost of the rooms are R: 70000
```

```
The sales tax paid is : 10%
The total cost per booking is R77000
```

   Submit a printout of the program and output.

Question 2b

Four experiments are performed, each consisting of five test results. The results for each experiment are given in the following list. Write a program using a nested loop to compute and display the average of the test results for each experiment. Display the average with a precision of two digits after the decimal point.

1st experiment results:	23.2	31	16.9	27	25.4
2nd experiment results:	34.8	45.2	27.9	36.8	33.4
3rd experiment results:	19.4	16.8	10.2	20.8	18.9
4th experiment results:	36.9	39	49.2	45.1	42.7

Use the input provided in the given list and execute the program.

   Submit a printout of the program and output.

Question 2c

In this program, you have to make use of the switch statement.

The average life expectancy (in hours) of a lightbulb based on the bulb's wattage is listed in the table below:

Watts	Life expectancy (hours)
25	25000
40	1000
60	1000
75	750
100	750

Write a program that when given a bulb's wattage, displays the average life expectancy.

   Submit a printout of the program and output.

Question 3:

The Golden Sales Company pays it salespeople R12.50 for each item they sell. Given the number of items sold by a salesperson, your program should first print a heading, then calculate, and print the amount of pay due.

A function named `printHeading` (with no parameters) displays the following message:

```
**************************************************
                GOLDEN SALES COMPANY
This program inputs the number of items sold by a
Salesperson and prints the amount of pay due.
**************************************************
```

Another function, `calculatePay` displays the amount pay due to a salesperson. The function multiples the numer of items sold with 12.50 to compute the pay to be payed out. The function has one value parameter items representing the number of items sold by a salesperson.

A main program inputs an integer value (`items`). It displays the description of the program by calling the function `printHeading`. The program then calls the function `calculatePay` to calculate and display the amount of pay due.

Sample run:

```
**************************************************
                GOLDEN SALES COMPANY
This program inputs the number of items sold by a
Salesperson and prints the amount of pay due.
**************************************************
Please input the number of items sold
125
The amount pay due is R 1562.50
```

Question 4a

Write the functions `printHeading` and `calculatePay` as well as the main program.

submit   Submit a printout of the program and output.

Question 4b

Change the program so that the pay amount is displayed in the main program instead of function `calculatePay`.

submit   Submit a printout of the program and the output.

Question 5:

Question 5a

Write a function named `integerPower()` that accepts two integer numbers (`base` and `exponent`) as formal parameters and returns the value of base ^{exponent}. For example,

```
integerPower(3,4) = 3 * 3 * 3 * 3 = 81
```

The function `integerPower()` should use a `for` or `while` loop to do the calculation. (Do not use any math library functions.)

Include the `integerPower()` function in a working program. Again, the `main()` function should input the values, correctly call `integerPower()` and display the value returned by the function, all with appropriate messages.

submit   Submit a printout of the program and output.

Question 5b

Write a function named `isEqual()` that accepts two `char` values as formal parameters and returns `TRUE` if the characters ar the same otherwise `FALSE`.

Include the `isEqual()` function in a working program. The `main()` function should input the values, correctly call `isEqual()` and display the value returned by the function, all with appropriate messages.

submit   Submit a printout of the program and output.

Question 5c

Write a function named `twice()` that accepts two `integer` values as formal parameters. The function then multiplies each parameter with 2 which is returned to the calling program.

Include the `twice()` function in a working program. The `main()` function should input the values, correctly call `twice()` and display the values returned by the function, all with appropriate messages.

submit   Submit a printout of the program and output.

Question 6:

A particular talent competition has 5 judges, each of whom awards a score between 0 and 10 to each performer. Fractional scores, such as 8.3, are allowed. A performer's final score is determined by dropping the highest and lowest score received, then averaging the 3 remaining scores. Write a program that uses this method to calculate a contestant's score. It should include the following functions:

a) A function that asks the user for a judge's score, store it in a reference parameter variable, and validate it. This function should be called by main once for each of the 5 judges. Do not accept judge scores lower than 0 or higher than 10.
b) A function to calculate and display the average of the 3 scores that remain after dropping the highest and lowest scores the performer received. This function should be called just once by main, and should be passed the 5 scores. The two functions, described below, should be called by this function, which uses the returned information, to determine which of the scores to drop.
c) A function to find and return the lowest of the 5 scores passed to it.
d) A function to find and return the highest of the 5 scores passed to it.
e) A function to display the output.

The main function is given below. Implement the functions you have developed and execute you're the program using the following data:

12	24	23	4	15
12	4	15	8	59
y	Y	y	y	N

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

int main()
{
  double score1, score2, score3, score4, score5;
                                    // 5 judge's scores
  double finalScore;

  // Call getJudgeData once for each score to be input
  getJudgeData(score1);
  getJudgeData(score2);
  getJudgeData(score3);
  getJudgeData(score4);
  getJudgeData(score5);

  // Call calcScore to calculate the contestant's final score
```

```
        finalScore = calcScore(score1, score2, score3, score4,
score5);

        // Display output
        displayOutput(finalScore);
        return 0;

    }// end of main function
```

submit   Submit a printout of the functions and output.

End of Assignment 2

©
UNISA
2018