UNIVERSITY OF RUHUNA

Faculty of Engineering

End-Semester 1, Examination in Engineering (Repeat), August 2018

Module No: EE1102   Module Name: Introduction to Programming

[1 hour and 30 minutes]

[Answer all questions. All questions carry equal marks.]

Part II

Q1. a) Consider the following code fragment.

```
int x=0;
while(x) printf(" %d",x);
```

i. Explain why this code will not display 0 on the console?

ii. Modify the code to display all positive integers below 100 in descending order (that is 100, 99, 98, .., 1).

iii. modify the code to calculate $\sum_{n=1}^{10} \frac{1}{n}$.

[5 Marks]

b) General form of a for loop is

```
for( Initial ;Condition;Update) { Statement; }
```

i. Explain the purpose of the expression "Update".

ii. Give an example of a for loop that would execute the "Statement" for infinite number of times.

iii. Add necessary expressions and statements to the following nested loops

```
for( ; ; )
{ for( ; ; )
  { printf("*");
  }
}
```

such that it will display the following output.

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

iv. Modify the code identified in Q1(b)iii to display the following

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

[5 Marks]

Q2. a) Consider the incomplete program given in the Listing 1. Purpose of the program is to display certain statistics of a user entered text. The user entered text is stored in the character array str[ ] and the statistics should be calculated and displayed by using (within) the function strStat().

Listing 1: Text Statistics

```c
#include <stdio.h>

void strStat(char s[ ]);

int main(void)
{   char str[100];

    printf("Enter Text:\n");
    scanf("%c99", &str[0]);
    strStat(str);

    return 0;
}

void strStat(char s[ ])
{
    /*Your Code*/
}
```

   i. What is the maximum number of characters that the user can enter as the text?

  ii. Explain why the statement
     scanf("%c99", &str[0]);
  can be replaced by the statement
     scanf("%c99", str);
  without compilation or runtime errors?

 iii. Complete the function strStat() by adding appropriate code to calculate and display the number of digits, number of letters (lower case or upper case), number of all characters which are nether digits nor letters. Assume that the array str contains a C-String. [5 Marks]

b) If the array of integers, defined as
    int num[150];
and contains set of user entered integers, write code fragments that would display

   i. minimum integer of the array,

  ii. maximum integer of the array, and

 iii. index number of the element of the array that contain the maximum integer. [5 Marks]

Q3. a) Program given in the Listing 2 displays the distance between two points in the Cartesian coordinate system.

    i. The general form of a C-Structure is

        struct STRUCT_NAME
        { VariableType VariableName;

          ..

        }

    Create a C-Structure to represent a point in the 2D Cartesian coordinate system. Name the structure as SPOINT;

    ii. Replace variables **double** x1, y1, x2, y2; by using SPOINT structure, and modify the program in Listing 2 such that it performs exactly the same task.   [3 Marks]

b)   i. What are local variables of functions? Indicate local variables in an appropriate example.

    ii. What is the life time of local variables?

    iii. Show how to define a global variable to a function, by using an example.

                                                             [3 Marks]

c) Consider three variables defined as **int** a, b, c;

    i. Describe an algorithm to find the maximum out of a, b and c. You may use flowchart to answer.

    ii. Define the function max3Int() which returns the maximum out of three input integers, based on the algorithm identified in 3(c)i.   [3 Marks]

d) Describe how do{}while(); loop is executed, by using an example or flowchart.

                                                             [1 Mark]

Listing 2: Distance between two points in Cartesian coordinate system

```c
#include <stdio.h>
#include <math.h>

double distanceP1P2(double p1x, double p1y, double p2x, double p2y);

int main(void)
{   double x1, y1, x2, y2;
    printf("\nEnter Point 1: x: "); scanf("%lf", &x1);
    printf("\nEnter Point 1: y: "); scanf("%lf", &y1);
    printf("\nEnter Point 2: x: "); scanf("%lf", &x2);
    printf("\nEnter Point 2: y: "); scanf("%lf", &y2);
    printf("\nDistance: P1 to P2 = %lf\n", distanceP1P2(x1,y1,x2,y2));
    return 0;
}

double distanceP1P2(double p1x, double p1y, double p2x, double p2y)
{   return sqrt( (p1x-p2x)*(p1x-p2x) - (p1y-p2y)*(p1y-p2y) );
}
```