University of Ruhuna- Faculty of Technology
Bachelor of Information and Communication Technology Degree
Level 2 (Semester 1) Examination, November 2019

Course Unit: ICT2113- Data Structures and Algorithms   Time Allowed: **2 hours**

Answer all **four (04)** questions

This question paper contains **05 pages.**

Question 01

a) What are the major characteristics of an algorithm?

b)

    I.   What is time and space complexity of an algorithm?

    II.   Explain space complexity of following code sample.

```
int add (int a, int b)
{
        return a + b;
}
```

c) What are the three (03) differences between *bubble* sort and *insertion* sorting techniques?

d) Write a Function in C for Insertion sort, considering the algorithm given below.

```
insertionSort(array, size)
        mark first element as sorted
                for each unsorted element X
                'extract' the element X
                    for j <- lastSortedIndex down to 0
                        if current element j > X
                            move sorted element to the right by 1
                    break loop and insert X here
        end insertionSort
```

e) Consider the following array

        -12 6 9 13 21 34 58 88 92 106

Show all steps required to search for 34 in the above array using "Binary Search" method.

Question 02

a) Write down meanings of following queue operations.
   I.   Dequeue
   II.   Enqueue
   III.   Peek
   IV.   IsEmpty

b) Write down content of the queue after each operation.
   enqueue(5), enqueue(3), dequeue(), 🌀 enqueue(7), dequeue(), peek(), dequeue(),
   dequeue(), IsEmpty(), enqueue(9), enqueue(7), size(), enqueue(3), enqueue(5),
   dequeue().

c) What are the three (03) types of possible scenario of inserting an element to a circular queue?

d) Consider following algorithm which shows inserting an element to the circular queue.

   Step 1: if (rear+1)%max = front
            write " overflow "
            goto **step 4**
            [end of if]
   step 2: if front = -1 and rear = -1
            set front = rear = 0
            else if rear = max - 1 and front ! = 0
            set rear = 0
               else
               set rear = (rear + 1) % max
            [end of if]
   step 3: set queue[rear] = val
   step 4: exit

   Implement **insert (int item, int queue[])** function, converting the above algorithm into a C code.

e) Convert the following **postfix** expressions into equivalent **prefix** and **infix** forms.
   I.   A B + C * D E F G * + / + H -
   II.   A B * C + D / E F G + / -

   A B + * C D E F * G + / + H -

   A B + * C D E F * G / + + H -

   A + B * C·D E F * G / H ++ H -

   A + B * C + D + E * G

Question 03

a)

    I.   Name and describe three (03) types of link lists with suitable diagrams.

    II.   Write down two (02) advantages of circular link lists.

    III.   What are the key differences between arrays and link lists?

    IV.   Consider the following link list declaration. The function *createNode()* is written to create a new node of link list. Complete the blank lines (**A, B, C, and D**) with suitable C programming statements to create a new node of the list.

```
struct LinkedList{

 int data;

struct LinkedList *next;

};

typedef struct LinkedList *node;

node createNode(){

A // declare a node

B // allocate memory using malloc()

C // make next point to NULL

D //return the new node

}
```

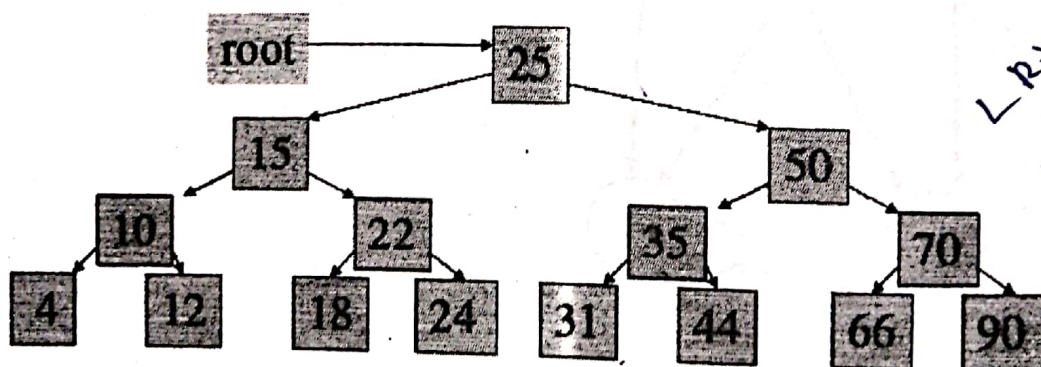b) I. Name three (03) types of binary tree traversal and write down traversal order of each of them

Diagram 1

L-Right-Root
L root right

Diagram 1

25 , 15 , 10, 4 ,12 , 22, 18 , 24 , 50,35,31,44,70,u
ac

II. Considering above binary tree (diagram 1), print the data in three (03) traversal orders which you have define in part (b) (I) above.

III. How many leaves are there in above tree?

IV. What is the height of node 35?

c) Draw binary trees for following expressions.

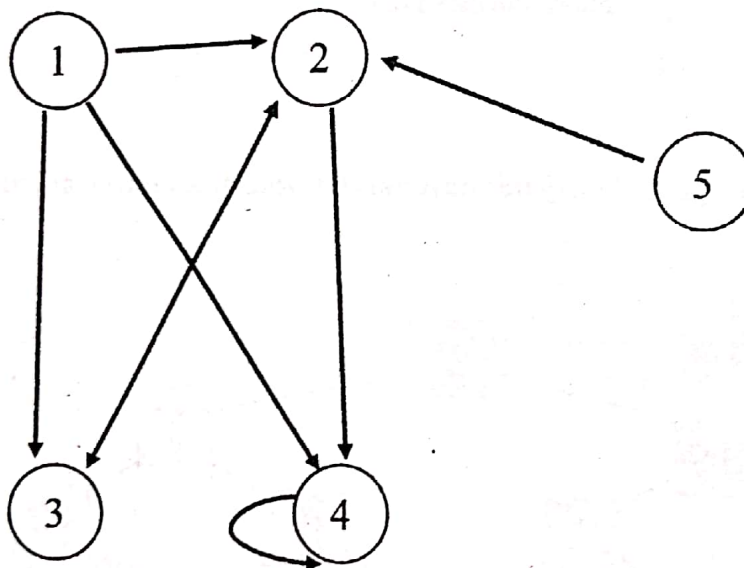I. $8 * (9 + 3) / 5 + (22 - 6)$

II. $(a + b) / ((a * b)) - c + d$

$$(a + b) / ((a * b) - c) + d$$

Question 04

a)

I. What are the basic properties of graph data structure?

II. Differentiate trees and graphs.

III. Explain following graph terminologies,
Adjacent Vertices, directed edge, undirected edge, parallel edges and self-loop.

b)

I. Write the adjacent matrix for the above graph.

II. Write the adjacency list for the above graph.

III. Write down three (03) differences between BFS (Breadth First Search) and DFS (Depth First Search) of a graph.

c)
   I. What is meant by recursion in computer programming?
   II. What are the key components of a recursive algorithm?
   III. Differentiate **recursion** and **iteration** based on given attributes of the following table.

Attributes	Recursion	Iteration
Structure		
Termination		
Speed		
Code size		
Memory use		

----------------------------End of the paper----------------------------