National University of Computer and Emerging Sciences, Lahore Campus

Course Name:	Data Structures and Algorithms	Course Code:	CS2015/ CS2002
Program:	BS Electrical Engineering	Semester:	Fall 2023
Duration:	3 Hours	Total Marks:	70
Exam Date:	23-Dec-2023	Weight:	50
Section:	All	Page(s):	10
Exam Type:	Final	CLO #	1, 2, 3, 4

Student Name:______________________________ **Roll No.**_______________ **Section:**________

Instructions:
1. Please answer all questions within the space provided.
2. Extra sheets should not be provided or collected. Last two pages are for rough work.
3. It is a closed book closed notes examination

Question No.1 (CLO1,PLO2,C4) **Marks: 10**

i. [Marks: 5] **Evaluate** the time complexity expression for the following code snippet and express it in terms of Big-Oh

```
for (i=1; i<=n; i=i*2){
    for (j=1; j <= i; ++j)
    cout<<j<<endl;
}
```

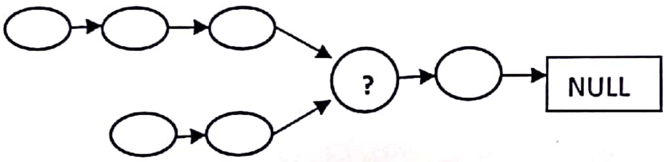
ii. [Marks: 05] **Analyze** the following code snippet for space complexity and express it in terms of Big-O. Show all the working for full marks. (Hint: assuming n is a multiple of 2)

```
int root(double n){
  if (n > 1.0)
  return(1 + root(n/2));
  else
        return 0;
  }
```

uestionNo.2 (CLO2, PLO3, C5) **Marks: 15**

i. [Marks: 05] **Generate** the prefix expression against the following postfix expression:
X Y Z / + E J % - K F * +

ii. [Marks: 10] Suppose there are two singly linked lists both of which intersect at some node (as shown below) and become the same list. The number of nodes in each list before they intersect is not known and both may have it different. List1 may have m nodes before it reaches intersection node and list2 may have n nodes where m and n may be m=n, m<n or m>n.



Construct an algorithm to find the merging node is as given below but its time complexity is O(mn).

```
 Node temp = first node of list1
 While(temp){
   Node temp2 = first node of list2
   While(temp2){
     If (temp == temp2)
       Break and return temp
     Else
       Temp2 = temp2->getNext();
  }
temp=temp->getNext();
}
```

Please give an algorithm that improves the performance of the above problem to $O(m+n)$.

QuestionNo.3 (CLO3, PLO3, C5) **Marks: 15**

A Graph class has the following declaration

```
class Graph{
private:
  const static int max_size = 10;
  int vertex_count = 0;

  string vertex_names[max_size];
  bool adjacency_matrix[max_size][max_size];

public:
  Graph(){
    for(int i=0;i<max_size;i=i+1)
      for(int j=0;j<max_size;j=j+1)
        adjacency_matrix[i][j] = false;
  }
  bool isConnected(void);
  bool hasCycles(void);
};
```

i. [Marks: 05] **Develop** the definition for **isConnected** function, which returns true if the graph is conn

ii. [Marks:10] **Develop** the definition for **hasCycles** function, which returns true if the graph has any cyclic path.

Question No.4 (CLO4, PLO5, C3) Marks: 39

A farmer has 8 mango trees in his garden, labeled "A" to "H". In the garden there is a designated fix path from on tree to the other. One day the farmer got sick as a result he could not push his loading cart, in which he used to collect his harvested mangoes, from one tree to the other. So, he decided to park his cart under one tree and fetch the harvested mangoes to the cart after picking them from each tree, one tree at a time. He asked his assistant to count the number of steps needed to walk from one tree to another. The assistant not being too savvy, brought the information in the following form.

F to A 60 steps.

C to H 20 steps.

F to D 20 steps

E to G 10 steps.

A to B 40 steps.

H to G 10 steps.

D to E 50 steps.

B to C 30 steps.

G to F 80 steps

E to C 15 steps.

i. [Marks: 05] Generate the adjacency matrix for the above graph.

	A	B	C	D	E	F	G	H
A								
B								
C								
D								
E								
F								
G								
H								

ii. [Marks: 10] Where should the farmer park his cart so that he needs to walk the least number of steps to harvest all the mango trees, once entering the garden. The door of the garden leads to tree "F" in 10 steps. Also write the path he must take for each tree.

Cart is stationed at: ________

Tree	Steps from Cart	Paths
A		
B		
C		
D		
E		
F		
G		
H		

i. [Marks: 05] Following code runs on an AVL Tree template class. **Construct** / draw the final representation of the tree.

```
AVLTree<int> A;
for(int i=1;i<100;i=i+10)
  A.insert(i)
```

iv. [Marks: 05] **Apply** quick sort algorithm to sort the array given below in descending order, using the first element of the array and subarrays as pivot.

6	1	5	7	9	10	8	2	4	3

Show the state of the array after the first recursive call of the quicksort function.

Show the state of the array after the second call of the quicksort function.

The quicksort function would be called how many times? ________

[Marks:5] License plate number of vehicles in Pakistan are alphanumeric of the form of LEA-1234. The first two letters represent the city of registration, and the remaining are incremented from A-0001 for each new vehicle registered. The traffic police want to implement the record in a hash table. The maximum size of hash table is 10 thousand records. **Write (construct)** the code for a hash function, which takes the number plate value as key of datatype string and returns an appropriate integer for data storage and retrieval.