USN:_____

Department of Artificial Intelligence and Machine Learning

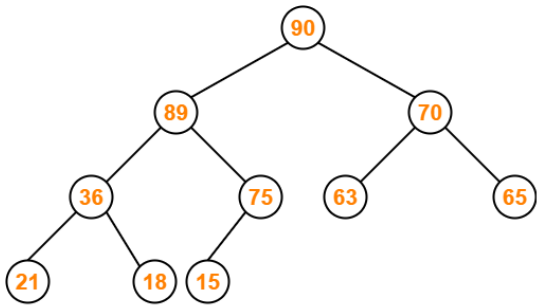
Course Code: 21AI33
Sem: III

Date: 18-Jan-2023
Duration: 90 Minutes

CIE-II

Data Structures and Data Analysis (DSDA)

Scheme and Solutions

SL. No		Questions	M	BT	CO
1	a	Given the following Max Heap, write various steps in deleting the data element 89 and also show heapifcation.  **ANS**: (1M x 4 Steps) Step 1: Search for the location of 89 Step 2: Swap 89 with the rightmost leave in the bottommost level, i.e., 15 Step 3: Compare 15 with the children, i.e., 36 and 75 Step 4: Swap 15 with the highest child, i.e., 75	04	03	01
	b	Write a C program to create a Max Heap of integers and display the same. **(Hint: Use arrays to represent the Max Heap)** **ANS: Scheme** Correct declaration of all variables and data types — 01 M Creation of Max Heap — 03 M Displaying the Max Heap and Overall coding effort — 02 M **ANS: Sample Solution** **# include <stdio.h>** **int arr[100],n;** **void display()** **{ int i;** **if(n==0)** **{** **printf("Heap is empty\n");** **return;** **}**	06	03	03

USN:_____

Department of Artificial Intelligence and Machine Learning

```
        for(i=0;i<n;i++)
                printf("%d ",arr[i]);
        printf("\n");
}/*End of display()*/

void insert(int num,int loc)
{
        int par;
        while(loc>0)
        {
                par=(loc-1)/2;
                if(num<=arr[par])
                {
                        arr[loc]=num;
                        return;
                }
                arr[loc]=arr[par];
                loc=par;
        }/*End of while*/
        arr[0]=num; /*assign num to the root node */
}/*End of insert()*/

main()
{
        int num;
        n=0;/*Represents number of nodes in the heap*/
        while(1)
        {
            printf("Enter the number to be inserted or -1 break the loop ");
                        scanf("%d",&num);
                         if (num == -1) break;
                        insert(num,n);
                        n=n+1;
         }
         display();
}
```

2 | Write a C program to create an adjacency matrix of a directed graph and display the graph using DFS. Assume maximum size of the adjancency mantrix A as A[MAXV][MAXV]. (**NOTE: Your program should accept vertices and edges as inputs.)** | 10 | 03 | 03

ANS: Scheme

Correct declaration of all variables and data types	02 M
Creation of an adjacency matrix for a graph	03 M
DFS logic	03 M
Output	02 M

ANS: Sample Solution

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

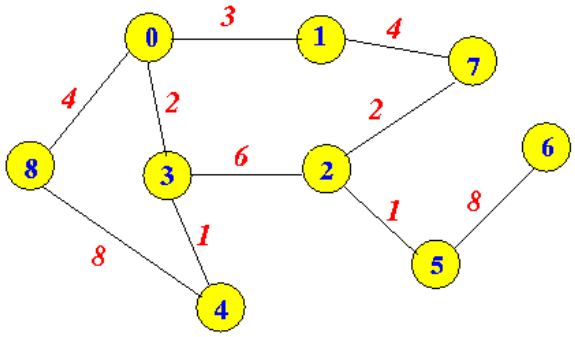
USN:_____

Department of Artificial Intelligence and Machine Learning

```
#include <stdlib.h>

int sourceV,Vertex,Edge,time,visited[10],Graph[10][10];

void DepthFirstSearch(int i)
{
   int j;
   visited[i]=1;
   printf(" %d->",i++);
   for(j=1;j<=Vertex;j++)
      {
      if(Graph[i][j]==1&&visited[j]==0)
      DepthFirstSearch(j);
      }
}

void main()
{
   int i,j,vertex1,vertex2;
   printf("\t\t\tGraphs\n");
   printf("Enter no. of edges:");
   scanf("%d",&Edge);
   printf("Enter no. of vertices:");
   scanf("%d",&Vertex);
   for(i=1;i<=Vertex;i++)
            {
            for(j=1;j<=Vertex;j++)
            Graph[i][j]=0;
            }
   for(i=1;i<=Edge;i++)
            {
            printf("Enter the edges in V1 V2 : ");
            scanf("%d%d",&vertex1,&vertex2);
            Graph[vertex1][vertex2]=1;
            }
   for(i=1;i<=Vertex;i++)
            {
            for(j=1;j<=Vertex;j++)
            printf(" %d ",Graph[i][j]);
            printf("\n");
            }
            printf("DFS Traversal\n ");
             scanf("%d",&sourceV);
            DepthFirstSearch(sourceV)
}
```

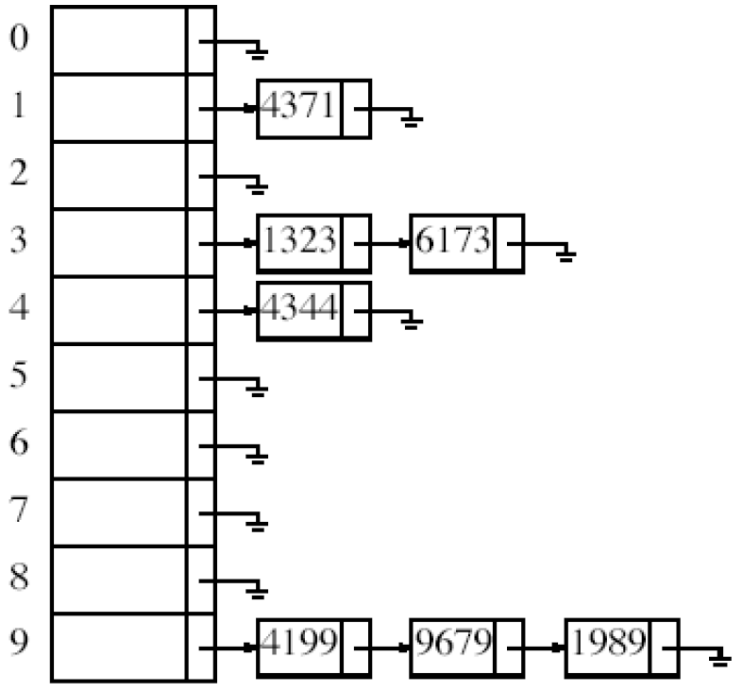
USN:_____

Department of Artificial Intelligence and Machine Learning

3	a	Write the DFS traversal for the given weighted graph. Assume starting vertex as 3.  **ANS:** Final Correct Answer: 3->4->8->0->1->7->2->5->6 ------ (2M) Traversal Steps based on least weight ------ (2M)	04	02	03

	b	Given the following C function, which is written for BFS traversal of a graph, with source vertex as v. Correct the logic if there are any errors, and write the corrected version. ***Assume adj[][], visited[], front, and rear have been properly initialised before the function was called.***
		
		void bfs(int v)
		{
		for (i = 1; i <= n; i++)
		if (adj[v][i] && visited[i])
		queue[rear++] = i;
		if (front <= rear)
		{
		visited[queue[front]] = 1;
		printf("Node visited = %d\n", queue[front]);
		bfs(adj[v][i]);
		}
		}
		
		ANS SAMPLE:
		
		void bfs(int v)
		{
		for (i = 1; i <= n; i++)
		if (adj[v][i] && !visited[i]) ------- 2M
		queue[++rear] = i; -------- 2M
		if (front <= rear)
		{
		visited[queue[front]] = 1;
		printf("Node visited = %d\n", queue[front]);
		bfs(queue[front++]); -------- 2M
		}
		}
			06	04	01

4	a	Give the interpretations of the following statements;	04	04	04

		1. H is universal if, for each x, y ∈ U, the number of h ∈ H such that h(x) = h(y) is at most |H|/m. **ANS**: Interpret the characteristic of hash functions belonging to the set H, where the conflicting hash function's ceiling limit should be |H|/m. 2. Searching and Deletion might take linear time in the case of Hash by Division. **ANS**: Interpret the nature of hash by division, which may force the uneven distribution of key values in various hash buckets. These two operations may not happen in constant time due to the list of key values in a single hash bucket.			
	b	Given the input {4371, 1323, 6173, 4199, 4344, 9679, 1989}, a fixed table size of 10, and a hash function h(x) = x mod 10, show the working of; **a. Hashing with Open addressing and b. Hashing with Quadratic probing** **ANS:** **a. Hashing with Open Addressing (03 M)**  **OR** **Open Addressing with Linear Probing**	06	03	02

USN:_____

Department of Artificial Intelligence and Machine Learning

5					

0	9679
1	4371
2	1989
3	1323
4	6173
5	4344
6	
7	
8	
9	4199

b. Hashing with Quadratic probing (03 M)

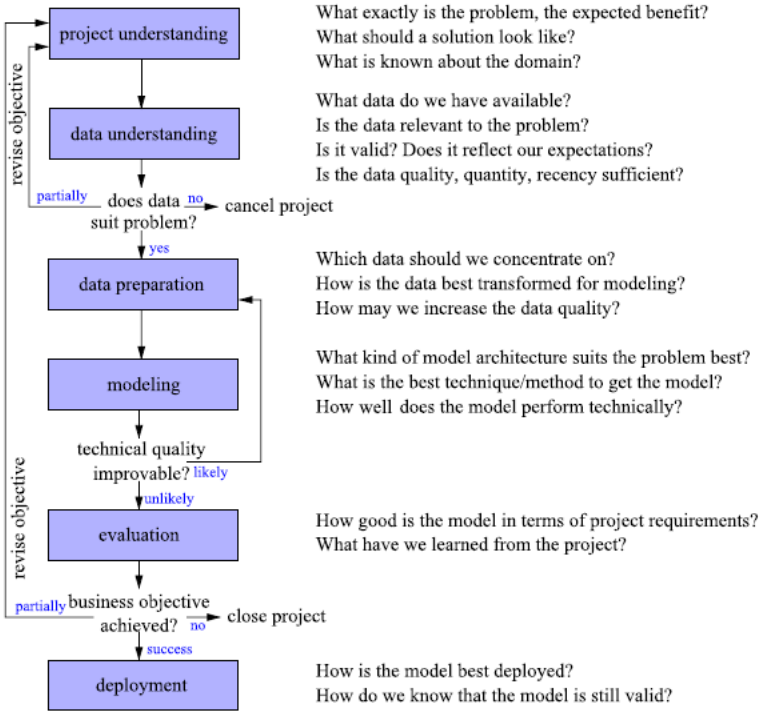
0	9679
1	4371
2	
3	1323
4	6173
5	4344
6	
7	
8	1989
9	4199

	Write a block diagram depicting various phases of the CRISP-DM Process and briefly explain the importance of all stages. **ANS**:	10	02	02

USN:_____

Department of Artificial Intelligence and Machine Learning

Block diagram (04M)
Explanation (06M)



Course Outcome	
CO1	Apply the knowledge of data structures in providing solutions to some software development requirements.
CO2	Perform data analysis of some real-world scientific/business use cases and present the analysis results.
CO3	Investigate appropriate data structures and understand requirements in solving some problems of industry and society.
CO4	Use data analysis tools to illustrate the principles of data interpretation, statistical analysis, and graphical visualizations of the datasets.
CO5	Appraise data structures and analysis knowledge to build a successful career as an AIML engineer, work in teams, and communicate their ideas effectively.

M-Marks, BT-Blooms Taxonomy Levels, CO-Course Outcomes

Marks Distribution	**Particulars**	**CO1**	**CO2**	**CO3**	**CO4**	**CO5**	**L1**	**L2**	**L3**	**L4**	**L5**
	Max Marks	10	16	20	04	--	--	14	26	10	--

USN:_____

Department of Artificial Intelligence and Machine Learning

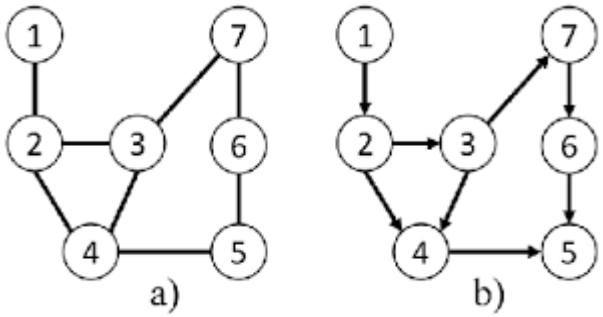
QUIZ-II

Course Code: 21AI33 **Date:** 18-Jan-2023

Sem:III **Duration:** 20 Minutes

Data Structures and Data Analysis (DSDA)

Scheme and Solutions

SL. No	Questions	M	BT	CO
1	Differentiate Data and Knowledge. Give an example for each. **ANS:** Any valid difference (1M) + Examples for each (01M)	02	02	01
2	Write any two differences between Trees and Graphs. Give an application of a Graph. **ANS:** Any valid differences (1M) + Application of Graphs (1M)	02	01	03
3	Give Data Structure representations of the following graphs.  **ANS:** Adjacency Matrix/Adjacency list representation for each of the above(1M+1M)	02	02	01
4	How do Hash Tables are used to implement Set ADT? **ANS**: How hashing achieves Unique and Membership properties of a set (1M + 1M)	02	02	01
5	Is the hash function **h(x)=x mod m** a perfect hash function? Justify. **ANS:** No, the reason is single hash value can be mapped to multiple key elements	02	02	01

Marks Distribution	Particulars	CO1	CO2	CO3	CO4	CO5	L1	L2	L3	L4	L5
	Max Marks	08	---	02	--	--	02	08	--	--	--