MODEL ANSWER
B.Tech. (Third Semester) Examination 2013-14
(Information Technology)
Data Structure and Programming Methodology (IT2104N)
Time Allowed: Three hours Maximum Marks: 60

……

Note: The paper consists of two sections. Section A and Section B. Section A is compulsory. In Section B attempt any one question from each unit.

Section A

Q.1 - Objective type Questions. (10×1=10)

a. The Average case occur in linear search algorithm
   1. When Item is somewhere in the middle of the array
   2. When Item is not in the array at all
   3. When Item is the last element in the array
   4. When Item is the last element in the array or is not there at all

   Answer - 1 - When Item is somewhere in the middle of the array

b. On which principle does stack work?
   1. FILO
   2. FIFO
   3. LILO
   4. Both a and c above.

   Answer - 1 - FILO

c. Which of the following ways below is a pre order traversal?
   1. Root->left sub tree-> right sub tree
   2. Root->right sub tree-> left sub tree
   3. right sub tree-> left sub tree->Root
   4. left sub tree-> right sub tree->Root

   Answer - 1 – Root->left sub tree-> right sub tree

d. Which of the following linked list below have last node of the list pointing to the first node?
   1. Circular doubly linked list
   2. Circular linked list
   3. Circular singly linked list
   4. Doubly linked list

   Answer – 3 - Circular singly linked list

e. In _____tree, the heights of the two child sub trees of any node differ by at most one
   1. Binary tree
   2. Red black tree
   3. Splay tree
   4. AVL Tree

   Answer – 4 – AVL Tree

f. Worst time & average time complexity of quick sort is –
   1. O (n^2), O (n log n)
   2. O (n log n), O (n log n)
   3. O (n^2), O (n^2)
   4. O (n log n), O (n^2)

   Answer – 1. O (n^2), O (n log n)

g. ……..Is not an attribute for an entity EMPLOYEE?
   1. Name
   2. Date of birth
   3. Sex
   4. All are attributes.

   Answer – 4. All are attributes.

h. Which of the following is not a type of operation on data structure?
   1. Traversing
   2. Searching
   3. sorting
   4. Counting

   Answer – 4 - Counting

i. Worst time complexity of merge sort is ……**O(nlogn)**……

j. If S_1 = HEL & S_2 = LO then output of $S_1//S_2$ operation is ……**HELLO**……..

Q.2 - Short answer Type. (5×2=10)

a. What is binary search tree?

Ans – A binary tree T is called BST if each node N has the following property –
The value at N is greater than every value in the left subtree of N and is less than every value in the right sub tree of N.

b. What is non linear data structure?

Ans – A data structure in which elements are not accessed in sequential manner. Ex. Tree, Graphs etc.

c. What is AVL tree?

Ans - In AVL tree, the heights of the two child sub trees of any node differ by at most one and valid balance factors are 0, 1 and -1 only.

d. What are the various operations that can be performed over data structure?

Ans – Traversing, Searching, Sorting, Insertion, Deletion.

e. What are the different types of rotation in AVL?

Ans – LL, RR, LR. RL

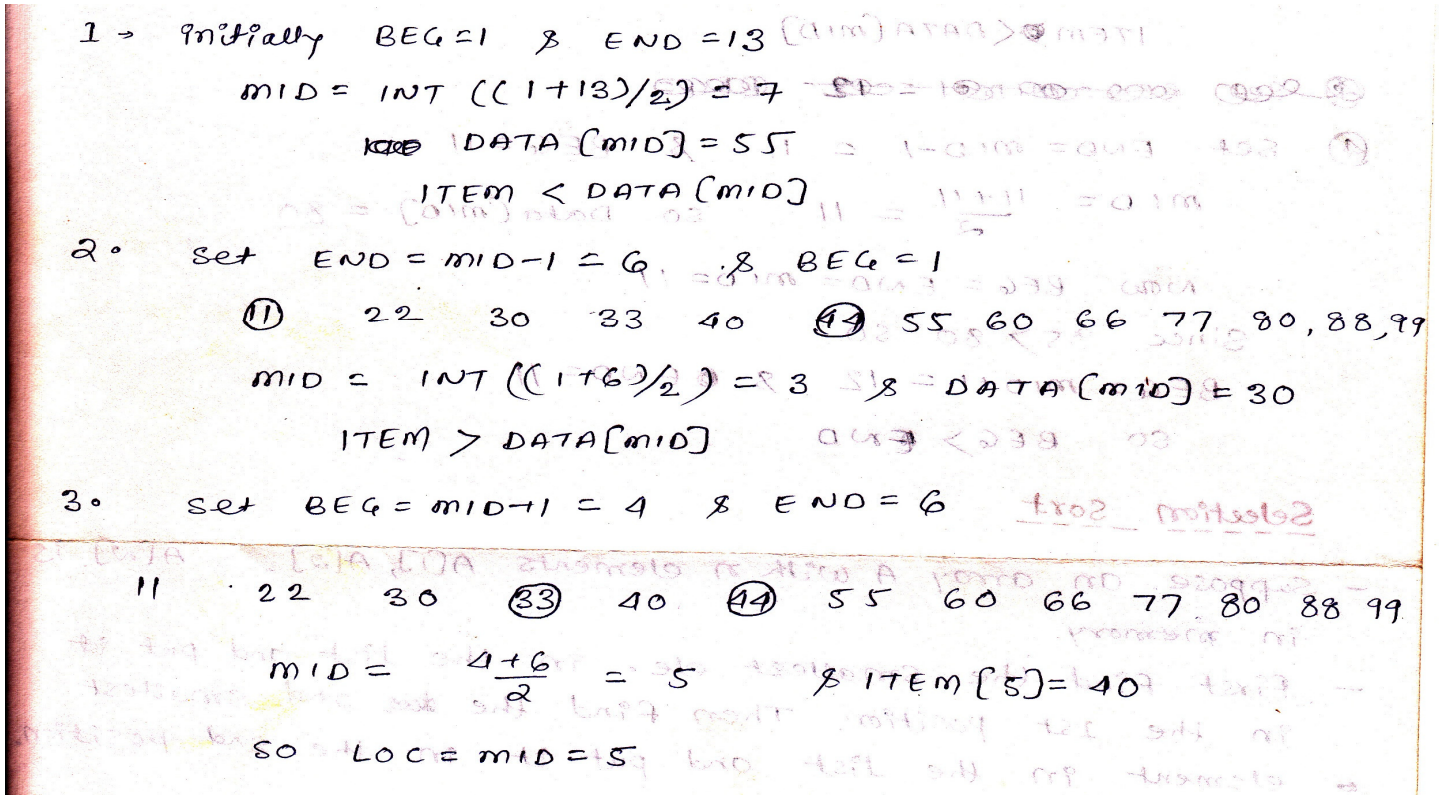
Section B

Note: Attempt any one question from each unit. Each question carries 8 marks.

Unit 1

Q. 3- Given sorted 13 elements array DATA – 11, 22, 30, 33, 40, 44, 55, 60, 66, 77, 80, 88, 99. Apply binary search for ITEM = 40. What is the location of ITEM? [8]

Ans -



OR

Q. 4 - Explain the Merging operation in details? Explain with the help of example? [8]

Ans –

Merging :- (Combining)

- Suppose A is a sorted list with r elements and B is a sorted list with s elements. merging combines the elements of A and B into a single sorted list C with n = r+s elements is called merging.
- we must always keep track of the locations of the smallest element of A and the smallest element of B Let NA & NB denote these locations respectively. also let PTR denote the location in C to be filled.
- so initially we set NA = 1, NB = 1 and PTR = 1. at each step of algo. we compare A[NA] and B[NB] & assign the smaller element to C[PTR].
- Then we increment PTR by setting PTR = PTR+1 and we either increment NA by setting NA = NA+1 or increment NB = NB+1 accerding to whether the new element in C has come from A or from B.
- If NA > r then remaining ele. of B are assigned to C or if NB > s then the remaining elements of A are assigned to C.

Ex:- ① A = ↓NA 1 3 5 7 B = ↓NB 2 4 6 8, 9, 10

C = ↓PTR 1, 2, 3, 4, 5 6 7 ; 8, 9, 10

② A = 2, 4, 6, 8, 9, 10 B = 1, 3, 5, 7

C = 1, 2, 3, 4, 5, 6, 7, ; 8, 9, 10

③ A = 1, 2, 3 B = 4, 5, 6, 7

C = 1, 2, 3 ; 4, 5, 6, 7 ....

(28)

Q. 5 - Write an algorithm for linked list implementation of stack operation (Push & Pop Operation). [8]

algo:- PUSH_LINKSTACK (INFO, LINK, TOP, AVAIL, ITEM)

1. If AVAIL = NULL then write OVERFLOW & Exit
2. Set NEW = AVAIL & AVAIL = LINK[AVAIL]
3. Set INFO[NEW] = ITEM
4. set LINK[NEW] = TOP
5. set TOP = NEW
6. Exit.

algo :- POP_LINK STACK (INFO, LINK, TOP, AVAIL, ITEM).

1. If TOP = NULL the write UNDERFLOW & Exit
2. Set ITEM = INFO[TOP]
3. Set TEMP = TOP & TOP = LINK[TOP]
4. Set LINK[TEMP] = AVAIL, & AVAIL = TEMP
5. Exit.

OR

Q. 6 - Write an algorithm to find whether a particular element is present or not in a linked list if the list is sorted. [8]

Ans – If Link List is sorted in ascending order -

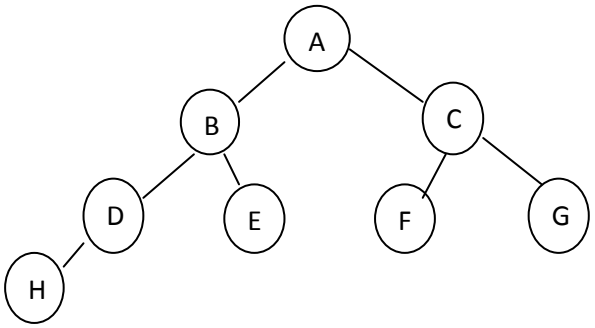
1. Set PTR = Start.
2. Repeat step 3 while PTR != Null
3. If ITEM > INFO [PTR]
   Set PTR = LINK [PTR]
   Else if ITEM = INFO [PTR]
   Set LOC = PTR & Exit.
   Else
   Set LOC = NULL & Exit.
4. Set LOC = NULL
5. Exit.

If Link List is sorted in descending order -

1. Set PTR = Start.
2. Repeat step 3 while PTR != Null
3. If ITEM < INFO [PTR]
   Set PTR = LINK [PTR]
   Else if ITEM = INFO [PTR]
   Set LOC = PTR & Exit.
   Else
   Set LOC = NULL & Exit.
4. Set LOC = NULL
5. Exit.

Unit 3

Q. 7 – If the given Binary tree T is [8]



Find the preorder, in order and post order traversal of Binary Tree T.

Ans –

Pre – ABDHECFG

In – HDBEAFCG

Post - HDEBFGCA

OR

Q. 8 – Define the following terminologies – Root, Successor, Predecessor, edge, path, height, Level, leaf node. [8]

Ans – Root – First node of any Tree is root node.

Sucessor – For any node N the root of left subtree of N is called left successor and the root of right subtree of N is called right successor.

Predecessor – Every node N in BT except the root has a unique parent called the predecessor of N.

Edge – connection between any 2 node is edge.

Path – sequences of edges.

Height – total no. of levels + 1 is height.

Level – each node in BT is assigned a level no. the root R of the tree T is assigned the level no. 0.

Leaf node – a terminal node is called leaf node.

Unit 4

Q. 9 - Explain Depth First Search with suitable example? [8]

Ans –

Depth first search (DFS) :-

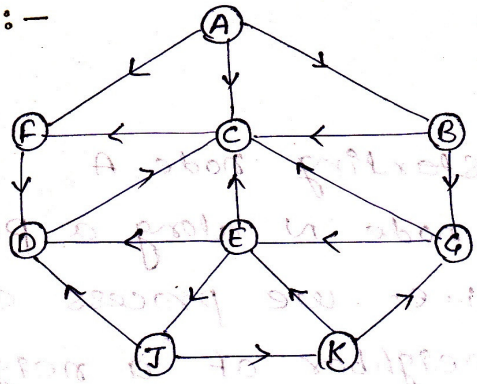
General idea :-

- First we examine the starting node A
- Then we examine each node N along a path P which begins at A i.e. we process a neighbor of A, then a neighbor of a neighbor of A & so on.
- after coming to a "dead end" i.e. to the end of the path P we backtrack on P until we continue along another path P' & so on.
- In DFS we use stack.

algo: (DFS) :-

1. Initialize all nodes to ready state. (St. 1)
2. Push the starting node A onto stack & change its status to the waiting state (St. 2)
3. Repeat step 4 & 5 until STACK is empty.
4. Pop the top node N of STACK. Process N & change its status to the processed status (St. 3)
5. Push onto STACK all the neighbors of N that are still in the ready state (St. 1) & change their status to the waiting state (St. 2)
6. Exit.

Que:-



Find & print all the nodes reachable from the node J (including J itself). Use DFS?

(a) initially push J onto the STACK

STACK : J

(b) Pop & print the top ele. J & then push onto the stack all the neighbors of J (that are in the ready state) as follows:

Print J STACK : D, K

(c) Pop & print top ele K & push all neigh. of K (ready state)

Print : K STACK : D, E, G

(d) Print G STACK : D, E, C

(e) Print C STACK : D, E, F

(f) Print F STACK : D, E,

(g) Print E STACK : D

(h) Print D STACK :

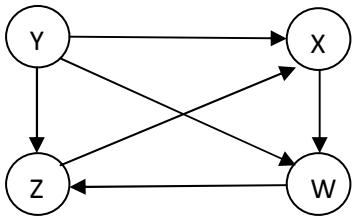
Stack is now empty, so that DFS of G starting at J is now completed

The nodes which were printed

J, K, G, C, F, E, D are the nodes which are reachable from J.

OR

Q. 10 – What is Adjacency Matrix? Draw the Adjacency Matrix of the following graph – [8]



Ans –

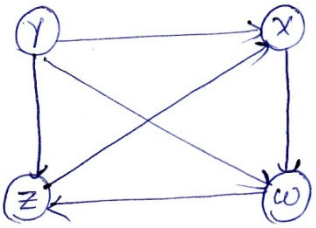
adjacency matrix :-

Suppose G is a simple directed graph with m nodes & suppose $v_1, v_2 \ldots v_m$ are nodes. Then the adjacency matrix $A = (a_{ij})$ of the graph G is the $m \times m$ matrix defined as follows:

$$a_{ij} = \begin{cases} 1 & \text{if } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise} \end{cases}$$

matrix which contains only 0 & 1 is called a bit matrix or Boolean matrix.

adjacency matrix of following graph:-



	X	Y	Z	W
X	0	0	0	1
Y	1	0	1	1
Z	1	0	0	0
W	0	0	1	0

Unit 5

Q. 11 - Explain in brief indexed sequential file organization? [8]

Ans – File organization is the methodology which is applied to structured computer files. Files contain computer records which can be documents or information which is stored in a certain way for later retrieval. File organization refers primarily to the logical arrangement of data (which can itself be organized in a system of records with correlation between the fields/columns) in a file system. It should not be confused with the physical storage of the file in some types of storage media. There are certain basic types of computer file, which can include files stored as blocks of data and streams of data, where the information streams out of the file while it is being read until the end of the file is encountered.

We will look at two components of file organization here:

1. The way the internal file structure is arranged and
2. The external file as it is presented to the O/S or program that calls it. Here we will also examine the concept of file extensions.

We will examine various ways that files can be stored and organized. Files are presented to the application as a stream of bytes and then an EOF (end of file) condition.

Techniques of File Organization

The three techniques of file organization are

1. Sequential
2. Indexed Sequential
3. Direct file organization

OR

Q. 12 - How to store & manage data in a file system? [8]

Ans – In this question very briefly explain data storage process and management and explain any one method of storage management.