Course Code : CST 252 / CDT 201 **OSPT/MW – 21 / 1508**

Third Semester B. Tech. / B. E. (Computer Science and Engineering) / (Data Science) Examination

DATA STRUCTURES AND ALGORITHMS

Time : 2 Hours] [Max. Marks : 40

Instructions to Candidates :—

(1) Attempt all questions.
(2) All questions carry marks as indicated against them.
(3) Due credit will be given to neatness and adequate dimensions.

1. Consider 7×8×5 array, **X**, of 32–bit integers where X[3][3][3] is stored at the address location 3000 using C–style ordering.

   Estimate the base address of X and find the address of an element at X[2][5][4] using row–major and column–major addressing.

   Mention the expressions for the above ordering before computing the address locations. 5 (CO 1)

2. Transform the arithmetic expression, **A ^ (B / C) * (D – (E + B))** to its equivalent postfix form.

   Evaluate the obtained postfix expression for A = 2, B = 5, C = 2, D = 9 and E = 2.

   Show the contents (stack frame) at each stage of transformation and evaluation. 7 (CO 2)

3. Consider the doubly linked list and a node structure defined with "data" indicating information content of a node and "prev", "next" as list pointers referencing previous and next node of a linked list.

   For a linked doubly list pointed by **p**, develop the code for the functions whose signatures are mentioned below —

   (a) Add a node as last node in the list.

   ```
   struct node *insertEnd(struct node *p, int x) ;
   ```

   (b) Remove a second last node in the list and return information content of deleted node through **x**.

   ```
   struct node *delete2Last(struct node *p, int *x) ;
   ```

   8 (CO 2)

4. (a) Construct the binary tree for the following expression —

$$(a - b) * (c - d) / e * f$$

Print the in – order, pre – order and post – order traversals of the expression tree. 4 (CO 4)

(b) Consider the binary tree shown in Fig. 4b and state whether it is an AVL tree, justifying your answer.

If it is not an AVL tree, transform it into an AVL tree. Now, insert keys 10, 55 and 50 into the tree in mentioned order. Finally, remove the key 40 from the tree.

Show the step – by – step insertion / deletion by stating the AVL violations and appropriate rotation(s) required.

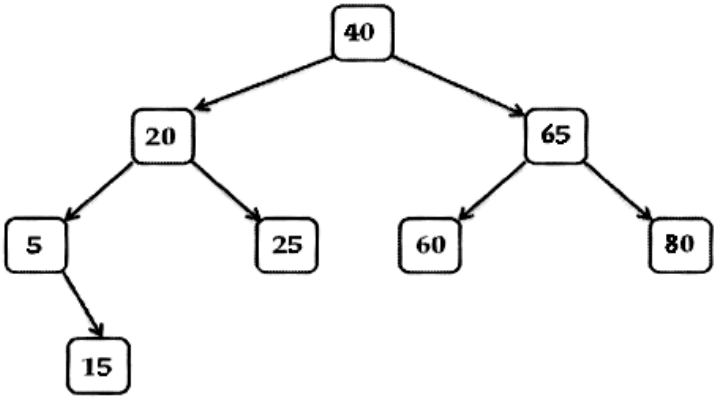

Fig. 4b 4 (CO 4)

5. A maxHeap is represented as an array —

Arr[ ] = {51, 32, 48, 25, 14, 40, 10, 17, 20}

Implement heap sort to order the list. Show intermediate heaps at each stage of the sorting process. 5 (CO 3)

6. (a) Consider the list of keys —

L = {453, 543, 908, 123, 782, 456, 276, 919, 234, 743}

and a hash function, h(k) = k % 11.

Show step by step construction of open addressing hash table using quadratic probing. 3 (CO 4)

(b) Consider the graph in Fig. 6b and apply Dijkstra's algorithm starting at vertex A. Note that the nodes will be processed in lexicographic sequence. Show contents of all data structures used during execution.

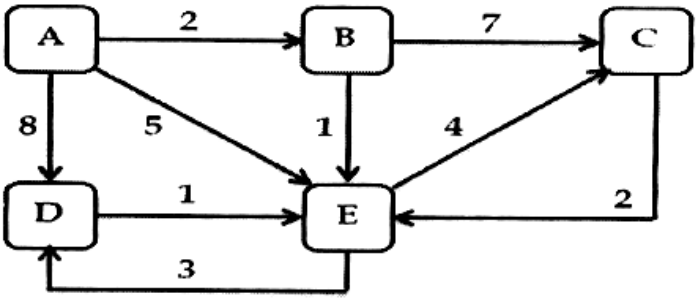

Fig. 6b 4 (CO 3)