53 (CS 304) DTST

2013

(December)

DATA STRUCTURES

Paper : CS-304

Full Marks : 100

Pass Marks : 30

Time : Three hours

The figures in the margin indicate full marks for the questions.

Answer ***any five*** *questions.*

1. *(a)* What is Big-op notation ? Write down the meanings of asymptotic time complexities $0(n)$, $0(2^n)$ and $0(\log n)$. Give examples of algorithms for each of them. 2+3+2=7

Or

What is asymptotic notation ? What are the different types of notation to express complexity of an algorithm. Define Time Complexity. 5+2=7

Contd.

(b) What do you mean by 2*D* array ? How can we represent 2*D* array in computer memory ? Consider an array *A* (3 × 4). Its base address is 200 and word per cell ($w = 2$). Find the address of *A* [2, 1] and *A* [2, 3] elements in all representations you have. 8

Or

Define relevant *C*-structures and union to describe an insurance company offering three kinds of policies-life, auto and home. Consider required fields of your choice wherever necessary.

(c) What is an Abstract Data Type (ADT). 5

2. Write algorithm/*C*-routines to do the following operations for a single linked list : 5×4=20

(i) Insert a new node to the front of the list.

(ii) Delete a node after the node pointed to by '*p*'.

(iii) Insert a new node after the node pointed to by '*p*'.

(iv) Traverse the list and return the sum of the integers contained in the list.

(v) Return the total no. of nodes contained in the list.

3. (a) What do you mean by Doubly Linked List ? What is the advantages of Doubly Linked List over the Singly Linked List. Define Header Linked List. 4+2=6

Or

Write Push and Pop C-subroutines for a stack being implemented as a linked list. 3+3=6

(b) What is an AVG tree ? How do AVG trees overcome the disadvantages of Binary Search Tree (BST) ? Create an AVG tree starting with initially empty tree with elements in order

30, 20, 10, 40, 50, 60, 25, 70, 90, 55, 15, 18, 14. 3+4+7=14

4. (a) Explain why a Queue is called FIFO structure. Represent pictorially a Queue in the form of a linked list. 2+2=4

(b) Consider an empty Queue in the form of an array. Show the status of Front and Rear indices if following tasks are performed. 1×6=6

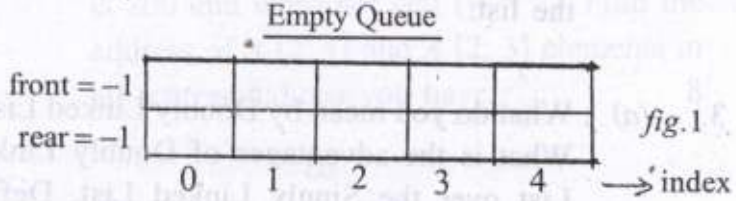

*fig.*1

(i) A, B and then *C* inserted, *(ii) A* deleted, *(iii) D* and then *E* inserted, *(iv) B* and *C* deleted, *(v) F* inserted, *(vi) D* deleted.

(c) Consider the following graph : 10

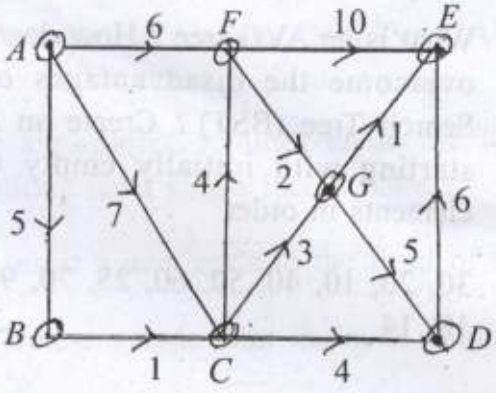

fig. 2

Find out the shortest path. Starting from node A to all other nodes using Dijkstra's algorithm.

5. *(a)* Sort the following sequence of elements using Quick Sort technique. What is the average time complexity of Quick Sort ? 65, 70, 75, 80, 85, 60, 55, 50, 45 8

(b) Compare time complexities of Selection sort and Quick Sort techniques. Write an algorithm/*C*-subroutine for Bubble Sort technique. 3+6=9

(c) Describe how does Hashing improve the efficiency of search technique ? What is collision in hashing technique ? 2+1=3

6. *(a)* Following are the inorder and preorder traversal of a Binary Tree.

Preorder : G, B, Q, A, C, K, F, P, D, E, R, H

Inorder : Q, B, K, C, F, A, G, P, E, D, H, R

Draw the diagram of the tree. 6

(b) What is binary search tree (BST) ? Consider the following BST : 4

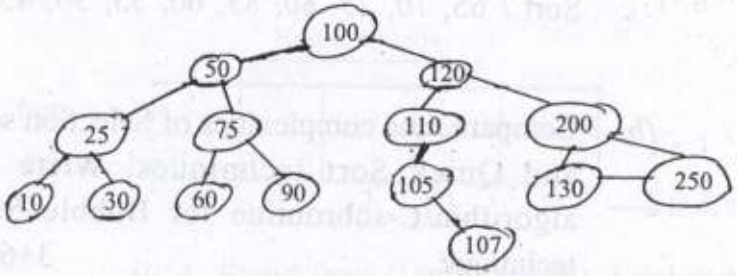

fig. 3

Show the sequence to find the elements 107 and 250.

(c) Evaluate the following arithmetic expression using stack : 10

$6+2\uparrow 3+9/3-4*5$

Or

How will you resolve Hash Clashes by Open Addressing ? How will you eliminate primary clustering and secondary clustering ? 3+7=10

7. *(a)* Define order of a *B*-tree. Consider the *B*-tree of order 5, shown in the following figure. 6

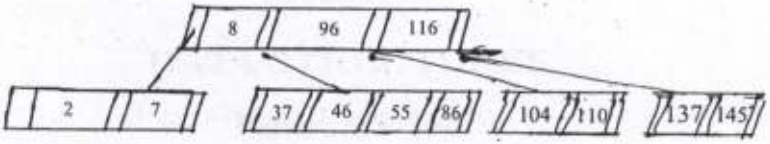

fig. 4

Show the status of the tree after inserting elements 4, 5, 58, 6 in order.

(b) Differentiate between DFS and BFS with example. 4

(c) Write algorithm/C-subroutines for insert and delete operations for a Queue when implemented as a linked list. 5+5=10