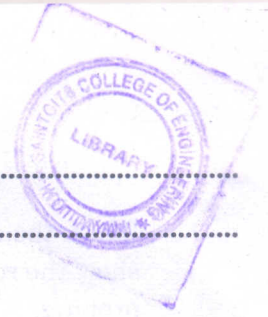


G 1528

(Pages : 2)

Reg. No.....

Name.....



B.TECH. DEGREE EXAMINATION, MAY 2015

Fourth Semester

Branch : Computer Science and Engineering / Information Technology

CS 010 403/IT 010 405—DATA STRUCTURES AND ALGORITHMS (CS, IT)

(New Scheme—2010 Admission onwards)

[Regular/Improvement/Supplementary]

Time : Three Hours

Maximum : 100 Marks

Part A

Answer all questions.

Each question carries 3 marks.

1. What is hashing ?
2. Define Stack.
3. Briefly explain about garbage collection.
4. Define tree traversal and mention the type of traversals.
5. What is insertion sort ? How many passes are required for the elements to be sorted ?

(5 × 3 = 15 marks)

Part B

Answer all questions.

Each question carries 5 marks.

6. Define collision ? Explain collision-resolution techniques.
7. What is queue ? Write an algorithm to implement queue with example.
8. Write a routine to insert an element in a linked list.
9. Formulate an algorithm to search an element in a binary tree.
10. Discuss any one sorting method with example.

(5 × 5 = 25 marks)

Part C

Answer all questions.

Each question carries 12 marks.

11. How will you resolve the collisions while inserting elements into the hash table using separate chaining and linear probing ? Write the routines for inserting, searching and removing elements from the hash table using the above mentioned techniques.

Or

Turn over

12. Given the input {4371, 1323, 6173, 4199, 4344, 9679, 1989} and a hash function of $h(x) = x \pmod{10}$ show the resulting : (a) Separate chaining hash table ; (b) Open addressing hash table using linear probing.
13. How a queue does work ? Explain the algorithm for inserting and deleting from a queue.

Or

14. Write a C program to perform the following stack operations. Declare a structure with one integer data type. Add necessary variables to ensure that you can create a linked representation. (a) Write a function called 'pop' that takes two parameters: an integer pointer and a stack from which we need to pop this element. (b) A display function that would print the contents of stack. Add necessary functions to check for emptiness or fullness of the stack ?
15. Formulate an algorithm to insert an element in a doubly linked list.

Or

16. Write algorithm to delete an element from anywhere in a doubly linked list. An element is a structure variable that contains an integer data field and a string data field.
17. Write the routines to insert and remove a node from Binary Search Tree.

Or

18. Write the procedures to implement single and double rotations while inserting nodes in an AVL tree.
19. Write down the complete quicksort algorithm and illustrate its working to sort the list (45, 23, 11, 35, 62, 87, 24, 66) ?

Or

20. Write down the complete heap sort algorithm and illustrate its working to sort the list (25, 73, 10, 95, 68, 82, 22, 60) ?

(5 × 12 = 60 marks)

