

**Code No: NR-21003-MCA**

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD**

**MCA-I Semester Regular Examinations, February 2010**

**DATA STRUCTURES**

**Time: 3hours**

**Max.Marks:60**

**Answer any Five questions**

**All questions carry equal marks**

---

- 1.a) What are the properties of linear and non linear data structures. Give example data structures for each
- b) Write a C program to find the transpose ( $A^T$ ) of a given matrix A.
- 2.a) Write a C-program to create a singly linked list and display the elements.
- b) Explain with an example how the linked lists are helpful for polynomial representation.
- 3.a) Write a C-program to evaluate an expression written in post fix notation, using stack data structure
- b) Illustrate with stack simulation how the factorial (n) is computed using recursion when n=3.
- 4.a) Explain how the circular queue is defined. Explain how the front and rear pointers are updated for
  - i) initialization
  - ii) insertion and
  - iii) deletion are performed.
- b) What is priority queue? Explain its implementation with an example
- 5.a) Which sorting technique is typically followed by an invigilator in the examination hall who sorts the student answer scripts based on student roll numbers. Explain the steps in this algorithm.
- b) Explain how the radix sort is implemented for the following integer indices.  
45, 5, 340, 92, 89, 18, 6, 422, 1333, 51
- 6.a) Write the recursive C-program to implement binary search. What is its time complexity?
- b) Explain the characteristics of Hashing techniques. Explain any one hashing technique
- 7.a) Construct
  - i) Binary search tree and
  - ii) heap tree for the following index values  
55, 33, 42, 9, 75, 65, 14, 2, 59, 94
- b) Write an algorithm for non-recursive in order tree traversal.
- 8.a) Define height-balanced tree. Explain the rules followed to make a binary search tree as height balanced tree
- b) Explain how the inorder tree traversal is simplified with the help of threads. Illustrate with an example.