Register Number.....

B.E. DEGREE EXAMINATIONS: APRIL /MAY 2009

Third Semester

ELECTRICAL AND ELECTRONICS ENGINEERING

U07EE305 Data Structures and Algorithms

Time: Three Hours **Maximum Marks: 100**

Answer ALL the Questions:-

PART A (20 × 1 = 20 Marks)

1. The tool used to specify the logical properties of a data type is called
   A. Abstract Data Type B. Absolute Data Type
   C. Attribute Data Type D. Additional Data Type

2. The Data structure in C that allows overlaying of different members is _____
   A. Union B. Structure C. Array D. List

3. Function calling itself is called
   A. Iteration B. Recursion C. Repetition D. Reversion

4. The address of a[i][j] in row major ordering is
   A. Base+(i-1)*no. of cols + j B. Base + i*no. of cols. + j
   C. Base+(i-1)*j D. Base+(i-1)*(j-1)

5. Push down list is nothing but a
   A. Queue B. stack C. Linked List D. Array

6. The data structure in which deletion is allowed only on the largest item is
   A. Priority queue B. Descending Priority Queue
   C. Ascending Priority Queue D. Queue

7. _____ is an ordered set of items from which items may be deleted at either end
   A. Enqueue B. Deque C. Queue D. Stack

8. An illegal attempt to remove an element from an empty queue is called
   A. Pop B. Overflow C. Underflow D. Push

9. A strictly binary tree with n leaves always contains _____ nodes
   A. 2n-1 B. 2^n-1 C. 2n D. 2n+1

10. Pre order traversal is also known as
   A. Symmetric order B. Depth First Order C. Linear order D. Post order

11. Ordered set of ordered tree is
   A. Tree set B. Spanning Tree C. Cluster D. Forest

12. A node with no sub tree is

A. Root B. Leaf C. Sibling D. Node

13. Quick sort is also called as

A. Partition Exchange Sort B. Decision Sort C. Prime Sort D. Radix Sort

14. Average sorting time for binary tree sort is

A. O(logn) B. O(n logn) C. O(log2n) D. O(n)

15. _____ is based on the values of the actual digits in the positional representation of the numbers being sorted

A. Radix sort B. Merge Sort C. Selection sort D. Bubble sort

16. A balanced binary tree is also called

A. Trie Structure B. Search Tree C. AVL Tree D. Rotation Tree

17. A path from node to itself is

A. sink B. Route C. Edge D. Cycle

18. Ford_Fulkerson algorithm is used to solve

A. Congestion problem B . Flow problem C. Ring problem D. Cycle problem

19. Round robin algorithm requires only O(e log log n) operations if _____ are used.

A. Queue B. Priority Queue C. Stack D. Circular Queue

20. Graphs can be represented using

A. Adjacency matrix B. List C. Stack D. Index

PART B (5 x 16 = 80 Marks)

21. a. (i) Explain in detail with an example how structures are implemented in C. (8)

(ii) Write a recursive function that solves towers of Hanoi problem and discuss about the efficiency of recursion. (8)

(OR)

21.b. Discuss in detail about array ADT and explain how one and two dimensional arrays are implemented in C.

22. a. (i) Briefly discuss about implementation of priority queues in C with suitable example (10)

(ii) Explain linked implementation of Queues. (6)

(OR)

22. b. (i) Explain the advantages of circular linked list over linked list with an example. (10)

(ii) Briefly explain about linked implementation of stack. (6)

23. a. (i) Discuss in detail various operations on binary search trees. (12)

(ii) Draw an expression tree for the expression 3+4*(6-11)/5+3 and evaluate the same. (4)

(OR)

23. b. (i) Explain clearly about threaded binary tree, its implementation and traversal of the same. (10)

(ii) What are the possible representations of binary tree? Explain (6)

24. a. (i) Sort the following using Quick sort mechanism. (8)

25, 57, 48, 37, 12, 92, 86, 33

(ii) Explain how binary search is advantageous compared to sequential search with an example. (8)

(OR)

24.b. Discuss in detail the heap sort with an example.

25. a. (i) Briefly explain digkstra's algorithm with an example (10)

(ii) Explain how graphs can be represented with linked list. (6)

(OR)

25. b. (i) Discuss in detail depth first traversal and its applications and efficiency (8)

(ii) Briefly explain the Prim's algorithm (8)

****