No of Pages : 3 **Course Code : 15XW23**

Roll No:
(To be filled in by the candidate)

PSG COLLEGE OF TECHNOLOGY, COIMBATORE - 641 004
SEMESTER EXAMINATIONS, SEPTEMBER / OCTOBER 2018
MSc – SOFTWARE SYSTEMS Semester : 2
15XW23 DATA STRUCTURES AND ALGORITHMS

Time : 3 Hours **Maximum Marks : 100**

INSTRUCTIONS:
1. Answer **ALL** questions. Each question carries 20 Marks.
2. Subdivision (a) carries 3 marks each, subdivision (b) carries 7 marks each and subdivision (c) carries 10 marks each.

1. a) Is the efficiency of algorithms dependent only on the input size? Are there any other factor that affects the efficiency? If so, mention the factors and justify how it affects the efficiency with an example.

b) (i) Find the time complexity for the following segment of the algorithm (3)

```
ALGORITHM (n)
    for (i=n; i >0; i/=2) {
      for (j=1; j<n; j*=2) {
        for (k=1; k<n; k+=2) {
          // Constant number of operations
} } }
```

(ii) You are given an array with positive integers. All integers occur even number of times except one. Write an algorithm to find this special integer in O(n) time complexity. (4)

c) Consider the tower of Hanoi puzzle in which there are *n* disks of different sizes that can slide onto any of three pegs. Initially, all the disks are on the first peg in order of size, the largest on the bottom and the smallest on top. The goal is to move all the disks to the third peg, using the second one as an auxiliary, if necessary. We can move only one disk at a time, and it is forbidden to place a larger disk on top of a smaller one. Therefore, to move the n^{th} disk, all other disks above it have to be moved to the auxiliary peg. Setup the recurrence relation for the above puzzle and derive its time complexity based on the basic operation.

2. a) Consider a two dimensional array 'A' of size n x n represented as a lower triangular matrix given below.

$$A = \begin{bmatrix} a_{11} & & & & \\ a_{21} & a_{22} & & & \\ a_{31} & a_{32} & a_{33} & & \\ .. & .. & .. & .. & \\ .. & .. & .. & .. & .. \\ a_{n1} & a_{n2} & .. & .. & a_{nn} \end{bmatrix}$$

The values in the matrix have to be stored in another linear array B with B[1] =a_{11}, B[2]=a_{21}, B[3]=a_{22}, B[3]=a_{31},…..a_{nn}. Each element occupies two bytes of memory. How many elements will Array B contain? Also, Compute the address of the last element in B assuming that the matrix A is of size 5 X 5 and the base address of B is 200.

b) (i) Obtain the array and multilist representation of the following sparse matrix. (3)

$$\begin{bmatrix} 0 & 0 & 5 & 1 \\ 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 3 \\ 0 & 0 & 4 & 0 \end{bmatrix}$$

(ii) A programming language permits indexing of arrays with character subscripts. The programming language uses ordinal number of the characters are indices. The ordinal number for A is 1, B is 2, C is 3 and so on. For Example, in array A['B':'F'], the starting index is 2 and the ending index is 6. Consider two arrays TEMP[1:5,-1:2] and CODE['A':'Z', -1:2] are stored in the memory starting from address 500. Also, CODE succeeds TEMP in storage. Calculate the address of : (I) CODE['N,2] (II) CODE['N',-1] (4)

c) Implement a stack S of n elements using arrays. Write procedures to perform PUSH and POP operations on the stack. Apply the procedures to convert an infix expression to a postfix expression. Trace the algorithm to find the postfix equivalent of a*(b↑c)*d+e↑f where ↑ represents exponentiation and evaluate the postfix expression obtained. Assume normal operator precedence.

3. a) What are the advantages of implementing a queue as a circular array rather than a linear array?

b) (i) Write a procedure to convert a linked stack in to a linked queue in such a way that the list is traversed only once. (3)

(ii) DQ is a output restricted dequeue implemented as a circular array. LEFT and RIGHT indicates the ends of the dequeue. INSERT(Element, [LEFT | RIGHT]) performs insertion of the specified Element to the left/right end and DELETE performs a delete operation. Execute the following instructions on the dequeue assuming that the maximum number of elements that DQ can hold is 6. Deletion can be performed only at the left end.

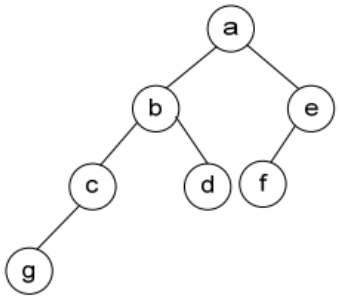
- INSERT(S5,LEFT)
- INSERT(K9,RIGHT)
- DELETE
- INSERT(V7,LEFT)
- INSERT(T5,RIGHT) (4)

c) Discuss the advantages and disadvantages of a circular linked list. Write procedures to insert and delete in a circular linked list. Write an algorithm to shift the elements of a singly linked list by one position towards the right. If the linked list has 12→56→34→40→7, then, after shift, the list is 7→12→56→34→40.

4. a) Construct a binary tree whose preorder traversal is K L N M P R Q S T and inorder traversal is N L K P R M S Q T.

b) (i) If a tree has n_1 nodes of degree 1, n_2 nodes of degree 2, …. n_m nodes of degree m, give a formula for the number of terminal nodes n_0 of the tree in terms of n_1, n_2, …., n_m. (3)

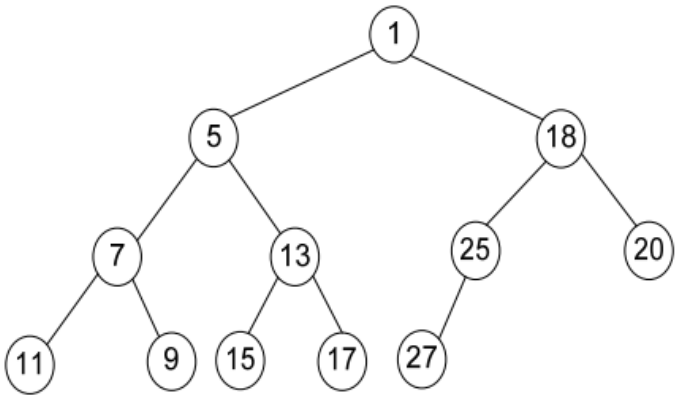
(ii) How are NULL pointers in a binary tree eliminated? Eliminate NULL pointers in the tree given below. (4)



c) (i) Write an algorithm to encode a given text using Huffman's algorithm. Apply the algorithm to encode the lyric of the rock song "Get a job. Shan na na na na na wah yip yip". Compute the total number of bits used to encode the words.

(OR)

(ii) Obtain an array and linked list representation of the binary tree given below. Which representation is efficient? Why? Write recursive procedures for tree traversals and traverse the tree given below using inorder, postorder and preorder traversals.



5. a) Sort the following elements using radix sort : 555,2,56,98,123,22

b) Write a procedure to sort a set of numbers using shell sort. Trace the algorithm on the list L= {7, 5, 5^2, 5^3, 5^4, 5^5, 5^6, 5^7, 5^9} for a sequence of increments {4,2,1}. The repeated occurrence of each element has been superscripted with their orders of occurrence.

c) Construct a hash table for the set of keys 85, 90, 70, 9, 56, 11, 71, 54, 10, 27, 35, 40, 64, 13, 31, 22, 77, 89, 53 using H(x) =x mod 9 as the hash function. Assume that the collisions are resolved using rehashing. Assume the rehashing function as H(k)=7- (k mod 7).

/END/

FD/RL