



B.E DEGREE EXAMINATIONS: NOV/DEC 2022

(Regulation 2018)

Third Semester

INFORMATION SCIENCE AND ENGINEERING

U18ISI3201 - Data Structures

COURSE OUTCOMES

CO1: Develop applications using stack and queue data structures

CO2: Develop applications to retrieve records from database using hashing techniques

CO3: Compare efficiency of various searching techniques using different tree data structures.

CO4: Compare efficiency of various sorting techniques using different data structures

Time: Three Hours

Maximum Marks: 100

Answer all the Questions

PART A (10 x 2 = 20 Marks)

(Answer not more than 40 words)

1. Calculate the worst case time complexity for the following operation. CO1 [K₃]

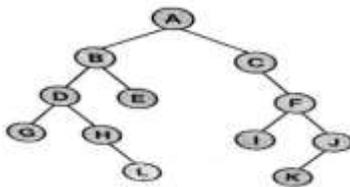
```
for (i=0;i<N;i++)
```

```
for (j=0;j<N;j++)
```

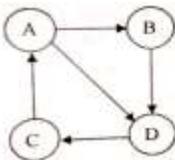
```
    k++;
```

2. Differentiate between Linear and Priority Queue CO1 [K₂]

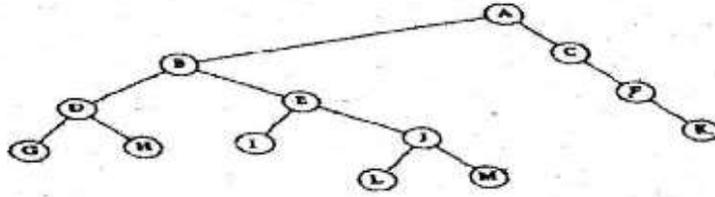
3. Traverse the tree and write the postorder traversal. CO3 [K₂]



4. Find the adjacency matrix for the following graph. CO3 [K₃]



5. For the given tree find the height and list the siblings for node E. CO3 [K₂]



- | | | | |
|-----|--|-----|-------------------|
| 6. | State the performance of quick sort in best, average and worst cases | CO4 | [K ₂] |
| 7. | Define secondary clustering | CO2 | [K ₂] |
| 8. | Write the formula for any one of the hash function | CO2 | [K ₂] |
| 9. | Compare the selection sort and merge sort | CO4 | [K ₂] |
| 10. | What are the properties of heap? | CO4 | [K ₂] |

Answer any FIVE Questions

PART B (5 x 16 = 80 Marks)

(Answer not more than 400 words)

- | | | | | | |
|-----|----|---|----|-----|-------------------|
| 11. | a) | Define Asymptotic notation. Discuss in detail about Big oh, Big omega and Big theta notation with neat graph. | 10 | CO1 | [K ₂] |
| | b) | Illustrate about binary search algorithm and analyze the time complexity | 6 | CO1 | [K ₃] |
| 12. | a) | Convert the following infix expression to postfix using stack. Depict with neat diagrams.
$A - (B / C + (D \% E * F) / G) * H$ | 8 | CO1 | [K ₃] |
| | b) | Construct the routine for enqueue() and dequeue() operations in queue using arrays with neat diagram. | 8 | CO1 | [K ₃] |
| 13. | a) | Develop the algorithm for singly linked list to insert a node after a given node, to find an element in the list and to check whether the list is empty or not | 8 | CO1 | [K ₃] |
| | b) | Write a routines for push() and pop() operations of stack using linked list with neat diagrams | 8 | CO1 | [K ₃] |
| 14. | a) | Discuss in detail about collision resolution techniques – separate chaining and open addressing with neat diagrams | 16 | CO2 | [K ₂] |
| 15. | a) | Describe about how to insert the element, find min and find max operations in Binary Search Tree with example program | 10 | CO3 | [K ₃] |
| | b) | Construct a B tree of order 5 for the following elements: 3, 14, 7, 1, 8, 5, 11, 17 | 6 | CO3 | [K ₃] |
| 16. | a) | Design an algorithm for insertion sort and apply the insertion sort for the following numbers: 13 26 1 2 27 38 15.
What is the running time for Insertion Sort | 8 | CO4 | [K ₃] |
| | b) | Design an algorithm and discuss in detail about Quicksort with an example. | 8 | CO4 | [K ₃] |
