

Sr No.	Question	CO
STACK		
1	Write an algorithm to reverse a string of characters using stack.	CO2
2	Convert $(A + B) * C - D \wedge E \wedge (F * G)$ infix expression into prefix format showing stack status after every step in tabular form.	CO2
3	List the applications of Stack.	CO2
4	Write an algorithm to reverse a string using stack.	CO2
5	Evaluate the following postfix expression using stack: (a) $9\ 3\ 4\ * \ 8\ + \ 4\ /\ -$ (b) $5\ 6\ 2\ + \ * \ 1\ 2\ 4\ /\ - \ +$	CO2
6	Arithmetic expression evaluation is an explanation of which data structure?	CO2
7	What is the reverse polish notation for infix expression $a / b * c$?	CO2
8	Write an algorithm to return the value of ith element from top of the stack.	CO2
9	Write an algorithm for inserting an element in a stack, removing an element from stack .	CO2
10	Write an algorithm to convert an infix expression to postfix expression. Show the working of the algorithm for the following expression. $A+B*C/D\$E-(F*G)$	CO2
11	Evaluate the following postfix expression using a stack.Show the steps. $2\ \$\ 3\ +\ 5\ * \ 2\ \$\ 2\ -\ 12\ \$\ 6$	CO2
12	Consider the stack S of characters, where S is allocated 8 memory cells. S: A,C,D, F, K, $_$, $_$, $_$	CO2
13	Describe the stack as the following operations take place. Pop(), Pop() ,Push(L), Push(P), Pop(), Push(R), Push (S), Pop()	CO2
14	Write a pseudocode for PUSH and POP operations of stack.	CO2
15	Write recursive algorithm to compute factorial of a given number. Which data structure can be used to implement this algorithm?	CO2
16	Evaluate the following postfix expression in tabular form showing stack after every step. $7\ 6\ +\ 4\ * \ 4\ 10\ +\ -\ 5\ +$	CO2
17	Write a C program to reverse a string using stack.	CO2
18	What is Stack? List out different operation of it and write algorithm for any two operation.	CO2
19	Write a 'C' program or an algorithm to convert infix expression without parenthesis to postfix expression.	CO2
20	Convert $A+(B*C-(D/E\wedge F)*G)$ infix expression into postfix format showing stack status after every step in tabular form.	CO2
21	List operations performed on a stack.	CO2
22	Write an algorithm to check if an expression has balanced parenthesis using stack.	CO2
23	What is postfix notation? What are its advantages? Convert the following infix expression to postfix. $A\$B-C*D+E\F/G	CO2
24	Write a C program to implement a stack with all necessary overflow and underflow checks using array.	CO2
25	Evaluate the following postfix expression using a stack.Show the stack contents. $AB*CD\$-EF/G/+$ $A=5, B=2, C=3, D=2, E=8, F=2, G=2$	CO2
26	Explain Recursion.	CO2
27	Write a pseudo-code for PUSH and POP operations of stack.	CO2
28	What is prefix notation? Convert the following infix expression into prefix. $A + B - C * D * E \$ F \$ G$	CO2
29	Convert the following infix expression into postfix. $A + B - C * D * E \$ F \$ G$	CO2

Sr No.	Question	CO
30	Enlist and briefly explain various applications of stack.	CO2
31	Write algorithms for Push and Pop operations on a stack.	CO2
32	What is Recursion? Write a pseudocode in 'C' language to find the multiplication of two natural numbers.	CO2
33	Differentiate between Stack and Queue.	CO2
34	Convert Infix Expression $A \wedge B * C - D + E / F / (G + H)$ into Postfix expression using stack.	CO2
35	Write an algorithms to convert Infix Expression(without parenthesis) into Postfix Expression	CO2
36	Evaluate the Postfix Expression $6\ 2\ 3\ +\ -\ 3\ 8\ 2\ /\ +\ * \ 2\ \$\ 3\ +$ using Stack.	CO2
37	Write algorithms for PUSH and POP stack operations.	CO2
38	Enlist applications of stack and queue.	CO2
39	Evaluate the following postfix expression using stack. Show each step. $5\ 3\ +\ 6\ 2\ /\ * \ 3\ 5\ * \ +$	CO2
40	Define following terms: (ii) recursion	CO2
41	Evaluate following expression using stack. $6\ 2\ 3\ +\ -\ 3\ 8\ 2\ /\ +\ * \ 2\ \$\ 3\ +$	CO2
42	Convert the following expression into postfix form. 1. $A \wedge B * C - D + E / F / (G + H)$ 2. $(A + B \wedge C \wedge D) * (E + F / D)$	CO2
43	Write a C program to implement stack using array with push, pop and display functions.	CO2
44	Write and discuss applications of stack and queue.	CO2
45	Convert following infix expressions to the postfix expressions. Shows stack trace. $A/B\$C+D*E/F-G+H\ (A+B)*D+E/(F+G*D)+C$	CO2
46	What do you mean by FIFO and LIFO?	CO2
47	What is Tower of Hanoi? Explain it with $n=3$.	CO2
48	What is Stack? Write down algorithms for performing POP and PEEP operations on a stack.	CO2
49	differentiate between stack & queue.	CO2
50	Distinguish between stack and queue.	CO2
51	What is top of stack? Why stack is called LIFO list?	CO2
52	What is stack? Explain operations on stack in detail.	CO2
53	Explain Tower Of Hanoi with example.	CO2
54	Evaluate the following postfix expression in tabular form: $3\ 5\ * \ 6\ 2\ /\ +$	CO2
55	Write a recursive function to compute factorial of a number. Show usage of STACK in recursionfor this function.	CO2
56	What is stack? Why do we use multiple stacks?	CO2
57	Convert the following infix expressions to their prefix and postfix equivalents 1. $A*B+C/D$ 2. $(A*B)+(C/D)-(D+E)$	CO2
58	Evaluate the following postfix expression using stack. $53+62/*35*+$	CO2
59	Explain following: (i) applications of stack	CO2
60	Write an algorithm for infix to postfix conversion.	CO2
61	Write an algorithm to evaluate postfix expression. Explain working of the algorithm using appropriate example.	CO2
62	Write a 'C' program to reverse a string using stack.	CO2
63	Describe: (1) Recursion (2) Tower of Hanoi	CO2

QUEUE

Sr No.	Question	CO
1	Compare: (2) Circular queue and Simple Queue.	CO2
2	Write an algorithm to implement insert and delete operations in a simple queue.	CO2
3	Describe: (2) Priority Queue	CO2
4	Write a 'C' functions to: (2) insert an element in circular queue.	CO2
5	Define priority queue.	CO2
6	Write an algorithm for simple queue with ENQUEUE operations.	CO2
7	Write a program to implement circular queue using array.	CO2
8	Explain the concept of circular queue. Compare circular queue with simple queue.	CO2
9	Explain double ended queue.	CO2
10	How many stacks are needed to implement a queue. Consider the situation where no other data structure like arrays, linked list is available.	CO2
11		CO2
	<p>Consider an example where the size of the queue is four elements. Initially the queue is empty. It is required to insert symbols 'A', 'B' and 'C'. delete 'A' and 'B' and insert 'D' and 'E'. Show the trace of the contents of the queue.</p>	
12	Write algorithm for inserting and deleting an element in circular queue.	CO2
13	Write and explain application of queue.	CO2
14	Discuss the variations of a queue.	CO2
15	Write a program to implement queue and check for boundary conditions.	CO2
16	Write algorithm for inserting an element in circular queue and deleting a node from a singly linked list.	CO2
17	Illustrate the working of priority queue with suitable example.	CO2
18	Write user defined C function for inserting an element into circular queue.	CO2
19	What is a Queue? Write down drawback of simple queue. Also write an algorithm for deleting an element from circular queue.	CO2
20	Mention variations of the queue data structure.	CO2
21	Write a C program to implement a circular queue using array with all necessary overflow and underflow checks.	CO2
22	Perform following operations in a circular queue of length 4 and give the Front, Rear and Size of the queue after each operation.	CO2
	1) Insert A, B	
	2) Insert C	
	3) Delete	
	4) Insert D	
	5) Insert E	
	6) Insert F	
	7) Delete	
23	Explain various applications of queue.	CO2
24	Explain Double-ended queue	CO2
25	Explain Priority queue	CO2
26	Write an algorithm to perform various operations (insert, delete and display) for simple queue.	CO2
27	Write differences between simple queue and circular queue.	CO2
28	Write an algorithm for insert and delete operations for circular queue.	CO2
29	Write algorithms for Insert and Delete operation in Circular Queue.	CO2
30	Explain Priority Queue?	CO2

Sr No.	Question	CO
31	Enlist applications of stack and queue.	CO2
32	Write a C functions for insertion and deletion operation in simple queue.	CO2
33	Write an algorithm to delete an element from circular queue. Show the steps of insertion and deletion operation in sample circular queue.	CO2
34	Explain with example: circular queue	CO2
35	Write and discuss applications of stack and queue.	CO2
36	Explain with example: priority queue	CO2
37	Write an algorithm to perform insert, delete and display operations for simple queue.	CO2
38	Explain double ended queue.	CO2
39	Explain the concept of circular queue. Compare circular queue with simple queue.	CO2
40	differentiate between stack & queue.	CO2
41	Explain Doubly linked list.	CO2
42	Write an algorithm to perform insert and delete operations on a circular Queue.	CO2
43	Discuss and write a program to implement queue functions using arrays.	CO2
44	Distinguish between stack and queue.	CO2
45	What is a circular queue? How do you check the queue full condition? Write an algorithm to count the nodes in a circular queue.	CO2
46	What is queue? Explain operations on queue in detail.	CO2
47	Explain advantages of circular queue over Simple queue.	CO2
48	Explain Dequeue and Priority queue in detail.	CO2
49	State disadvantages of simple queue. How to overcome it?	CO2
50	Write an algorithm for INSERT, DELETE and DISPLAY function of Circular Queue.	CO2
51	Write an algorithm to perform insert and delete operations on simple queue.	CO2
52	What is priority queue? Discuss its applications and implementation details.	CO2
53	Design an algorithm to perform insert operation in circular queue.	CO2
54	Write algorithm to (i) insert, and (ii) delete elements in circular queue.	CO2
LINKED LIST		
1	Write 'C' functions to: (1) insert a node at the end (2) delete a node from the beginning of a doubly linked list.	CO2
2	Compare: (1) Linked-list and Array.	CO2
3	Write a 'C' functions to: (1) insert a node at beginning in singly linked list.	CO2
4	Write 'C' structure of Singly linked list.	CO2
5	Write a program to implement stack using linked list.	CO2
6	Write 'C' functions to implement DELETE_FIRST_NODE and TRAVERSE operations in doubly linked list.	CO2
7	What does the following function do for a given Linked List with first node as head?	CO2
8	<pre>void fun1(struct node* head) { if(head == NULL) return; fun1(head->next); printf("%d ", head->data); }</pre>	CO2
9	Consider singly linked storage structures, Write an algorithm which inserts a node into a linked linear list in a stack like manner.	CO2
10	Write a program to implement a circularly linked list.	CO2
11	List the advantages of a doubly linked list over singly linked list.	CO2
12	Write an algorithm to swap two nodes, n and n+1, in a singly linked list.	CO2
13	What is a header node? Explain its importance.	CO2

Sr No.	Question	CO
14	Write an algorithm to count the number of nodes in a singly circularly linked list.	CO2
15	Write algorithm for inserting an element in circular queue and deleting a node from a singly linked list.	CO2
16	List the advantages of a doubly linked list over singly linked list.	CO2
17	Write down advantages of linked list over array and explain it in detail.	CO2
18	Write an algorithm to delete a node from doubly linked list.	CO2
19	What is the worst case time complexity of searching an element in a list? How?	CO2
20	Mention one operation for which use of doubly linked list is preferred over the singly linked list.	CO2
21	Write an algorithm/steps to traverse a singly linked list.	CO2
22	What is a header node and what is its use?	CO2
23	Write a program to insert and delete an element after a given node in a singly linked list.	CO2
24	Differentiate between arrays and linked list.	CO2
25	Create a doubly circularly linked list and write a function to traverse it.	CO2
26	Explain Circular linked list	CO2
27	Write algorithm(s) to perform INSERT_FIRST (to insert a node at the first position) and REVERSE_TRAVERSE (to display the data in nodes in reverse order) operations in doubly linked list.	CO2
28	Write a 'C' program to implement stack using linked list.	CO2
29	Write 'C' functions to implement INSERT_FIRST (to insert a node at the first position), DELETE_FIRST (to delete a node from the first position), DELETE_LAST (delete a node from the last position) and TRAVERSE (to display the data in nodes) operations in circular linked list.	CO2
30	Write an algorithm for insertion of node at last position in Liner Linked List.	CO2
31	Write an algorithm for deletion of node in Liner Linked List.	CO2
32	Write an algorithm for insertion of a node in Doubly Linked List.	CO2
33	Write an algorithm for deletion of a node in Doubly Linked List.	CO2
34	Describe the advantages of linked list over array.	CO2
35	Write an algorithm to insert a node at last position in doubly linked list.	CO2
36	Write an algorithm to print the singly linked list in reverse order using stack.	CO2
37	Write user defined C function to insert and delete a node at/from end in doubly linked list.	CO2
38	Differentiate between array and linked list.	CO2
39	Write user defined C function to insert and delete a node at/from specific location in singly linked list.	CO2
40	Discuss advantages and disadvantages of linked list over array.	CO2
41	Write a 'C' function to 1) deletion of a node at location 'LOC' from a singly linked list. 2) insert a node in beginning of a singly linked list.	CO2
42	Explain Doubly linked list.	CO2
43	Explain creation, insertion and deletion of doubly linked list with example.	CO2
44	Write and explain algorithm for deletion in Singly Linked List.	CO2
45	Write and Explain algorithm for insertion in doubly linked list.	CO2
46	Compare Array and Link list.	CO2
47	Write an algorithm for INSERT operation to insert a node at a given position in a Link list.	CO2
48	Write an algorithm to find length of a simple link list.	CO2
49	Write an algorithm to insert a node in a Circular Link List at the FIRST position.	CO2

Sr No.	Question	CO
50	Compare array and linked list.	CO2
51	Design an algorithm to merge two linked list.	CO2
52	Write user defined 'C' function to insert node at a specific location in singly linked list.	CO2
53	Write user defined 'C' function to delete node from end in circular linked list.	CO2
54	Write a 'C' program to implement queue using linked list.	CO2
55	Write user defined 'C' function to insert node at the end in circular linked list.	CO2
56	Write user defined 'C' function to delete node from a specific location in doubly linked list.	CO2
57	Write a 'C' program to implement stack using linked list.	CO2
ARRAY		
1	Explain Row major ordering and Column major ordering.	CO2
2	Explain Sparse matrix and its representation.	CO2