

Name of Subject	Data Structures	Subject Code	PCIT-101
Batch	2018 and onwards	Class	D2IT A and B

2 Marks Questions

1. Define data structures ?
2. Define algorithm ?
3. What do you mean by complexity of algorithm ?
4. Define asymptotic notation ?
5. What is timespace tradeoff ?
6. What are push and pop operations ?
7. What do you know about polish notation?
(HINT: Discuss polish and reverse polish notations)
8. What do you mean by dequeue i.e double ended queue ?
9. What is priority queue ?
10. What do you know about doubly linked list ?
11. Define circular linked list ?
12. What are AVL trees?
13. What do you know about adjacency matrix and adjacency test?
14. What is rehashing?
15. Define hashing function ?
16. Define double hashing?

5 Marks Questions

17. What are various asymptotic notations?
(HINT: big oh , theta , omega)
18. What do you mean by data structures? Discuss various types of data structures?
19. Discuss various operations that can be performed on data structures?
20. Write an algorithm to insert an element in an array ?
21. Write an algorithm to delete an element from an array ?

22. Write an algorithm to perform linear search ? Discuss its complexity ?
23. What are sparse matrices ? Discuss their storage ?
24. Implement stack using array ? Discuss its algorithm ?
25. Discuss quick sort algorithm as an application of stack ?
26. What do you mean by recursion? Explain how recursion is applied in the computation of factorial and fibonacci series ?
27. What do you mean by linear queue? What are its limitations? How circular queue overcomes its limitations ?
28. What do you know about linear linked list? Write an algorithm to insert and delete an element from linked list ?
29. What do you know about garbage collection and compaction ?
30. Write an algorithm for linked representation of stack ?
31. Write an algorithm for linked representation of queue ?
32. Demonstrate the sequential and linked representations of trees.
33. What do you know about preorder, inorder and post order traversals ?
34. Design an algorithm to insert and delete elements in a binary search tree?
35. What do you know about bubble sort algorithm? Discuss its complexity?

10 Marks Questions

36. What do you know about binary search? Write its algorithm? Discuss its complexity. Compare its performance with linear search ?
37. Write an algorithm to convert infix to postfix expression ?
38. Evaluate the given postfix expression

P : 60, 6, /, 5, 2, *, 5, - , +

Apply the algorithm clearly at each step.

[Answer:-

Scanned Elements	Operation	Stack Status
60	Push	60
6	Push	60, 6
/	Pop two operands from stack $60/6 = 10$ Push	10
5	Push	10, 5
2	Push	10, 5, 2
*	Pop two operands from stack $5*2 = 10$ Push	10, 10
5	Push	10, 10, 5
-	Pop two operands from stack $10 - 5 = 5$ Push	10, 5
+	Pop two operands from stack $10 + 5 = 15$ Push	15

39. Convert the following infix expression into postfix

Q: $A+(B*C-(D/E^F)*G)*H$

Apply the algorithm clearly at each step ?

[Answer:-

Symbol	Scanned	STACK	Postfix Expression	Description
1.		(Start
2.	A	(A	
3.	+	(+	A	
4.	((+(A	
5.	B	(+(AB	
6.	*	(+(*	AB	
7.	C	(+(*	ABC	
8.	-	(+(-	ABC*	'*' is at higher precedence than '-'
9.	((+(-(ABC*	
10.	D	(+(-(ABC*D	
11.	/	(+(-(/	ABC*D	
12.	E	(+(-(/	ABC*DE	
13.	^	(+(-(/^	ABC*DE	
14.	F	(+(-(/^	ABC*DEF	
15.)	(+(-	ABC*DEF^/	Pop from top on Stack, that's why '^' Come first
16.	*	(+(-*	ABC*DEF^/	
17.	G	(+(-*	ABC*DEF^/G	
18.)	(+	ABC*DEF^/G*-	Pop from top on Stack, that's why '^' Come first
19.	*	(+*	ABC*DEF^/G*-	
20.	H	(+*	ABC*DEF^/G*-H	
21.)	Empty	ABC*DEF^/G*-H*+	END

40. Write a recursive algorithm to solve the problem of Tower of Hanoi ?

41. Write an algorithm to insert and delete elements from circular queue? Discuss the complexity of these operations ?

(HINT: complexity is $O(1)$)

42. What do you mean by skip list? Write an algorithm to implement skip list and various operations on it?

43. Diagrammatically explain Breadth First Search (BFS) and Depth First Search (DFS).

44. Compare the various sorting algorithms. Discuss their performance by comparing their Big oh notations.

45. Discuss collision resolution techniques.
