# Data Structure MCQ

1) How can we describe an array in the best possible way?

1. The Array shows a hierarchical structure.
2. Arrays are immutable.
3. **Container that stores the elements of similar types**
4. The Array is not a data structure

2) Which of the following is the correct way of declaring an array?

1. **int javatpoint[10];**
2. int javatpoint;
3. javatpoint{20};
4. array javatpoint[10];

3) How can we initialize an array in C language?

1. int arr[2]=(10, 20)
2. int arr(2)={10, 20}
3. **int arr[2] = {10, 20}**
4. int arr(2) = (10, 20)

4) Which of the following is the advantage of the array data structure3Apple Watch Series 7

1. Elements of mixed data types can be stored.
2. Easier to access the elements in an array
3. Index of the first element starts from 1.
4. Elements of an array cannot be sorted

5) Which of the following highly uses the concept of an array?

1. Binary Search tree
2. Caching
3. Spatial locality
4. Scheduling of Processes

6) Which of the following is the disadvantage of the array?

1. Stack and Queue data structures can be implemented through an array.
2. Index of the first element in an array can be negative
3. Wastage of memory if the elements inserted in an array are lesser than the allocated size
4. Elements can be accessed sequentially.

7) What is the output of the below code?

1. #include <stdio.h>
2. **int** main()
3. {
4. **int** arr[5]={10,20,30,40,50};
5. printf("%d", arr[5]);
7. **return** 0;
8. }
9. Garbage value
10. 10
11. 50
12. None of the above

8) Which one of the following is the size of int arr[9] assuming that int is of 4 bytes?

1. 9
2. 36
3. 35
4. None of the above

9) Which one of the following is the process of inserting an element in the stack?

1. Insert
2. Add
3. Push
4. None of the above

10) When the user tries to delete the element from the empty stack then the condition is said to be a \_\_\_\_

1. Underflow
2. Garbage collection
3. Overflow
4. None of the above

11) If the size of the stack is 10 and we try to add the 11th element in the stack then the condition is known as\_\_\_

1. Underflow
2. Garbage collection
3. Overflow
4. None of the above

12) Which one of the following is not the application of the stack data structure

1. String reversal
2. Recursion
3. Backtracking
4. Asynchronous data transfer

13) Which data structure is mainly used for implementing the recursive algorithm?

1. Queue
2. Stack
3. Binary tree
4. Linked list

14) Which data structure is required to convert the infix to prefix notation?

1. Stack
2. Linked list
3. Binary tree
4. Queue

15) Which of the following is the infix expression?

1. A+B\*C
2. +A\*BC
3. ABC+\*
4. None of the above

16) Which of the following is the prefix form of A+B\*C?

1. A+(BC\*)
2. +AB\*C
3. ABC+\*
4. +A\*BC

17) Which of the following is not the correct statement for a stack data structure?

1. Arrays can be used to implement the stack
2. Stack follows FIFO
3. Elements are stored in a sequential manner
4. Top of the stack contains the last inserted element

18) If the elements '1', '2', '3' and '4' are added in a stack, so what would be the order for the removal?

1. 1234
2. 2134
3. 4321
4. None of the above

19) What is the outcome of the prefix expression +, -, \*, 3, 2, /, 8, 4, 1?

1. 12
2. 11
3. 5
4. 4

20) The minimum number of stacks required to implement a stack is \_\_

1. 1
2. 3
3. 2
4. 5

21) Which one of the following node is considered the top of the stack if the stack is implemented using the linked list?

1. First node
2. Second node
3. Last node
4. None of the above

22) Consider the following stack implemented using stack.

1. #define SIZE 11
2. **struct** STACK
3. {
4. **int** arr[SIZE];
5. **int** top=-1;
6. }

What would be the maximum value of the top that does not cause the overflow of the stack?

1. 8
2. 9
3. 11
4. 10

23) What is another name for the circular queue among the following options?

1. Square buffer
2. Rectangle buffer
3. Ring buffer
4. None of the above

24) If the elements '1', '2', '3' and '4' are inserted in a queue, what would be order for the removal?

1. 1234
2. 4321
3. 3241
4. None of the above

25) A list of elements in which enqueue operation takes place from one end, and dequeue operation takes place from one end is\_\_

1. Binary tree
2. Stack
3. Queue
4. Linked list

26) Which of the following principle does Queue use?

1. LIFO principle
2. FIFO principle
3. Linear tree
4. Ordered array

27) Which one of the following is not the type of the Queue?

1. Linear Queue
2. Circular Queue
3. Double ended Queue
4. Single ended Queue

28) Which one of the following is the overflow condition if linear queue is implemented using an array with a size MAX\_SIZE?

1. rear = front
2. rear = front+1
3. rear=MAX\_SIZE -1
4. rear = MAX\_SIZE

29) Which one of the following is the overflow condition if a circular queue is implemented using array having size MAX?

1. rear= MAX-1
2. rear=MAX
3. front=(rear+1) mod max
4. None of the above

30) The time complexity of enqueue operation in Queue is \_\_

1. O(1)
2. O(n)
3. O(logn)
4. O(nlogn)

31) Which of the following that determines the need for the Circular Queue?

1. Avoid wastage of memory
2. Access the Queue using priority
3. Follows the FIFO principle
4. None of the above

32) Which one of the following is the correct way to increment the rear end in a circular queue?

1. rear =rear+1
2. (rear+1) % max
3. (rear % max) + 1
4. None of the above

33) Consider the following code.

1. **int** fun()
2. {
3. **if**(isEmpty())
4. {
5. **return** -10;
6. }
7. **else**
8. {
9. **int** n;
10. n= q[front];
11. front++;
12. **return** n;
13. }
15. }

Which operation does the above code perform?

1. Enqueue
2. Dequeue
3. Return the front element
4. Both b and c

34) In the linked list implementation of queue, where will the new element be inserted?

1. At the middle position of the linked list
2. At the head position of the linked list
3. At the tail position of the linked list
4. None of the above

35) How many Queues are required to implement a Stack?

1. 3
2. 2
3. 1
4. 4

36) Which one of the following is not the application of the Queue data structure?

1. Resource shared between various systems
2. Data is transferred asynchronously
3. Load balancing
4. Balancing of symbols

37) Which of the following option is true if implementation of Queue is from the linked list?

1. In enqueue operation, new nodes are inserted from the beginning and in dequeue operation, nodes are removed from the end.
2. In enqueue operation, new nodes are inserted from the end and in dequeue operation, nodes are deleted from the beginning.
3. In enqueue operation, new nodes are inserted from the end and in dequeue operation, nodes are deleted from the end.
4. Both a and b

38) The necessary condition to be checked before deletion from the Queue is\_\_

1. Overflow
2. Underflow
3. Rear value
4. Front value

39) Which data structure is the best for implementing a priority queue?

1. Stack
2. Linked list
3. Array
4. Heap

40) Which of the following principle is used if two elements in the priority queue have the same priority?

1. LIFO
2. FIFO
3. Linear tree
4. None of the above

41) Which of the following statement is not true regarding the priority queue?

1. Processes with different priority can be easily handled
2. Easy to implement
3. Deletion is easier
4. None of the above

42) A linear data structure in which insertion and deletion operations can be performed from both the ends is\_\_\_

1. Queue
2. Deque
3. Priority queue
4. Circular queue

43) In the Deque implementation using singly linked list, what would be the time complexity of deleting an element from the rear end?

1. O(1)
2. O(n2)
3. O(n)
4. O(nlogn)

44) Which of the following data structure allows you to insert the elements from both the ends while deletion from only one end?

1. Input-restricted queue
2. Output-restricted queue
3. Priority queue
4. None of the above

45) What would be the output after performing the following operations in a Deque?

1. Insertfront(10);
2. Insertfront(20);
3. Insertrear(30);
4. Insertrear(40);
5. Deletefront();
6. Insertfront(50);
7. Deleterear();
8. Display();
9. 10, 20, 30
10. 50, 10, 30
11. 40, 20, 30
12. None of the above

46) In a circular queue implementation using array of size 5, the array index starts with 0 where front and rear values are 3 and 4 respectively. Determine the array index at which the insertion of the next element will take place.

1. 5
2. 0
3. 1
4. 2

47) If circular queue is implemented using array having size MAX\_SIZE in which array index starts with 0, front points to the first element in the queue, and rear points to the last element in the queue. Which one of the following conditions used to specify that the circular queue is empty?

1. Front=rear= -1
2. Front=rear=0
3. Front=rear+1
4. None of the above

48) Consider the implementation of the singly linked list having the head pointer only in the representation. Which of the following operations can be performed in O(1) time?

i)Deletion of the last node in the linked list  
ii)Insertion at the front of the linked list  
iii)Deletion of the first node in the linked list  
iv)Insertion at the end of the linked list

1. ii
2. both ii and iii
3. both i and iv
4. both i and ii

49) What would be the time complexity if user tries to insert the element at the end of the linked list (head pointer is known)?

1. O(1)
2. O(n)
3. O(logn)
4. O(nlogn)

50) Which of the following is the time complexity to search an element in the linked list?

1. O(1)
2. O(n)
3. O(logn)
4. O(nlogn)

51) Consider the following code

1. **struct** node
2. {
3. **int** data;
4. **struct** node \*next;
5. }
6. node ptr;

Which one of the following is the correct option to create a new node?

1. ptr= (node\*)malloc(sizeof(node\*))
2. ptr=(node)malloc(sizeof(node))
3. ptr=(node\*)malloc(sizeof(node))
4. None of the above

52) Which of the following statement is not true about the doubly linked list?

1. We can traverse in both the directions.
2. It requires extra space
3. Implementation of doubly linked list is easier than the singly linked list
4. It stores the addresses of the next and the previous node

53) What is the maximum number of children that a node can have in a binary tree?

1. 3
2. 1
3. 4
4. 2

54) Which one of the following techniques is not used in the Binary tree?

1. Randomized traversal
2. Preorder traversal
3. Postorder traversal
4. Inorder traversal

55) Which of the following options is not true about the Binary Search tree?

1. The value of the left child should be less than the root node
2. The value of the right child should be greater than the root node.
3. The left and right sub trees should also be a binary search tree
4. None of the above

56) How can we define a AVL tree?

1. A tree which is binary search tree and height balanced tree.
2. A tree which is a binary search tree but unbalanced tree.
3. A tree with utmost two children
4. A tree with utmost three children

57) Why do we prefer Red Black tree over AVL tree?

1. Red Black trees are not strictly balanced
2. Red black tree requires lesser rotations than AVL tree.
3. AVL tree needs more space to store the balance factor.
4. Both b and c

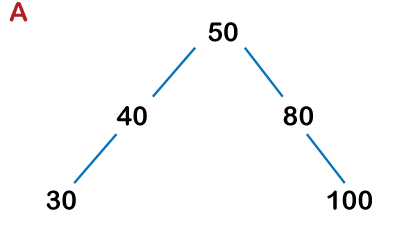
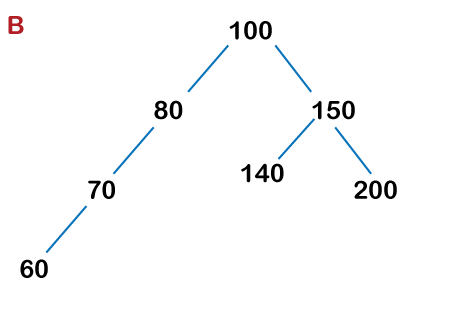
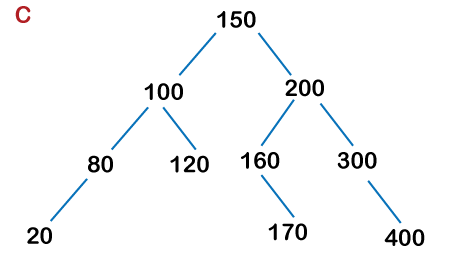
58) Which of the following satisfies the property of the Red Black tree?

1. A tree which is a binary search tree but not strictly balanced tree.
2. A node must be either Red or Black in color and root node must be black.
3. A tree with maximum three children
4. Both a and b

59) What would be the color of newly created node while inserting a new element in a Red black tree?

1. Black, if the new node is not a root node
2. Red, if the new node is not a root node
3. Black, if the new node is a root node
4. Both b and c

60) Identify the AVL tree among the following options?

1. A
2. C
3. Both A and C
4. B

**Answers**

1. C
2. A
3. C
4. B
5. C
6. C
7. A
8. B
9. C
10. A
11. C
12. D
13. B
14. A
15. A
16. D
17. B
18. C
19. C
20. 2
21. A
22. D
23. C
24. A
25. C
26. B
27. D
28. C
29. C
30. A
31. A
32. B
33. D
34. C
35. B
36. D
37. D
38. B
39. D
40. B
41. C
42. B
43. C
44. B
45. B
46. B
47. A
48. B
49. B
50. B
51. C
52. C
53. D
54. A
55. D
56. A
57. D
58. D
59. D
60. C