1. What is a stack?
   1. **A data structure that follows the LIFO (Last In, First Out) principle.**
   2. A data structure that follows the FIFO (First In, First Out) principle.
   3. A data structure that allows random access to elements.
   4. A data structure that stores elements in sorted order.
2. Which of the following operations is not performed on a stack?
   1. Push
   2. Pop
   3. **Insert**
   4. Peek
3. Which function is used to push an element onto the stack?
   1. **push()**
   2. insert()
   3. add()
   4. update()
4. What is the result of popping an element from an empty stack?
   1. **Stack Underflow**
   2. Stack Overflow
   3. Runtime Error
   4. No result, the stack remains unchanged
5. Which of the following data structures can be implemented using a stack?
   1. Queue
   2. Binary Tree
   3. Graph
   4. **All of the above**
6. The time complexity of push and pop operations on a stack is:
   1. **O(1)**
   2. O(n)
   3. O(log n)
   4. O(n log n)
7. Which data structure is typically used to implement function call stack?
   1. **Stack**
   2. Queue
   3. Array
   4. Linked List
8. In which order are elements stored in a stack?
   1. First In, Last Out (FILO)
   2. First In, First Out (FIFO)
   3. **Last In, First Out (LIFO)**
   4. Last In, Last Out (LILO)
9. The process of adding an element to a stack is called:
   1. Pop
   2. Peek
   3. **Push**
   4. Insert
10. Which of the following algorithms uses a stack for its implementation?
    1. **Depth-First Search (DFS)**
    2. Breadth-First Search (BFS)
    3. Dijkstra's Algorithm
    4. Prim's Algorithm
11. Which of the following is not a common application of a stack?
    1. Expression Evaluation
    2. Function Call Management
    3. Tree Traversal
    4. **Hashing**
12. The stack grows towards:
    1. Higher memory addresses
    2. **Lower memory addresses**
    3. Middle memory addresses
    4. The stack can grow in any direction
13. The operation of removing an element from a stack is called:
    1. **Pop**
    2. Peek
    3. Push
    4. Insert
14. Which of the following is not a stack implementation using an array?
    1. Static stack
    2. Dynamic stack
    3. Circular stack
    4. **Linked stack**
15. Which of the following is not a stack implementation using a linked list?
    1. Singly linked stack
    2. Doubly linked stack
    3. Circular linked stack
    4. **Array-based stack**
16. Which of the following is an advantage of using a linked list for stack implementation over an array?
    1. **Dynamic size**
    2. Constant time access
    3. Random access
    4. Sequential access
17. The process of accessing the top element of a stack without removing it is called:
    1. Pop
    2. **Peek**
    3. Push
    4. Insert
18. Which of the following is not a typical stack operation?
    1. isEmpty()
    2. isFull()
    3. **size()**
    4. clear()
19. The stack is an example of which type of data structure?
    1. **Linear**
    2. Non-linear
    3. Homogeneous
    4. Heterogeneous
20. Which of the following is not required for a stack implementation?
    1. Array or linked list
    2. Top variable to keep track of the top element
    3. Size variable to keep track of the number of elements
    4. **Front and rear variables**
21. Which of the following operations is performed in the evaluation of postfix expressions?
    1. Push
    2. Pop
    3. **Both Push and Pop**
    4. None of the above
22. Which of the following algorithms uses two stacks for its implementation?
    1. **Infix to Postfix Conversion**
    2. Postfix Expression Evaluation
    3. Infix Expression Evaluation
    4. Postfix to Infix Conversion
23. The time complexity of evaluating a postfix expression using a stack is:
    1. O(1)
    2. **O(n)**
    3. O(log n)
    4. O(n^2)
24. Which of the following is a valid postfix expression for the infix expression "4 + 5 \* 6"?
    1. **4 5 + 6 \***
    2. 4 5 6 + \*
    3. 4 5 + 6 \*
    4. 4 5 \* 6 +
25. Which of the following operations is not performed during the infix to postfix conversion?
    1. Push
    2. **Pop**
    3. Append to output
    4. Rearrange operators
26. The time complexity of infix to postfix conversion using a stack is:
    1. O(1)
    2. **O(n)**
    3. O(log n)
    4. O(n^2)
27. Which of the following is a valid infix expression for the postfix expression "5 3 + 2 \* 4 -"?
    1. 5 + 3 \* 2 - 4
    2. 3 + 5 \* 2 - 4
    3. **5 + 2 \* 3 - 4**
    4. 3 + 2 \* 5 - 4
28. The operation of rearranging operators during infix to postfix conversion is based on their:
    1. Precedence
    2. Associativity
    3. **Both Precedence and Associativity**
    4. None of the above
29. Which of the following is not required for infix to postfix conversion?
    1. Stack
    2. Input expression
    3. Output expression
    4. **Queue**
30. The reverse of a string can be obtained using a:
    1. Queue
    2. **Stack**
    3. Array
    4. Linked list
31. Which of the following is not a stack operation?
    1. Enqueue
    2. **Dequeue**
    3. Push
    4. Pop
32. Which of the following is not a valid stack implementation?
    1. Array of fixed size
    2. Array of dynamic size
    3. Linked list
    4. **Tree**
33. The size of the stack can be determined by:
    1. Counting the number of elements in the stack
    2. Checking if the stack is empty
    3. Checking if the stack is full
    4. **Both a) and c)**
34. Which of the following is not an application of stack in computer science?
    1. Function call management
    2. Expression conversion
    3. Memory allocation
    4. **Sorting algorithms**