

Q1. Which is the minimum number of stacks required to evaluate a fully parenthesized infix expression?

- a) 1
- b) 2
- c) 3
- d) 4

Answer: b

Q2. Which algorithm uses two stacks for expression evaluation?

- a) Dijkstra's Two-Stack Algorithm
- b) Shunting Yard Algorithm
- c) Bellman-Ford Algorithm
- d) Prim's Algorithm

Answer: a

Q3. Which of the following problems cannot be solved using only a stack?

- a) Balanced parentheses
- b) DFS traversal
- c) Shortest path in weighted graph
- d) Expression evaluation

Answer: c

Q4. Which stack operation causes stack overflow in recursive functions?

- a) Excessive pop
- b) Infinite recursion
- c) Shallow recursion
- d) None

Answer: b

Q5. What is the time complexity of reversing a stack using recursion?

- a) $O(1)$
- b) $O(n)$
- c) $O(n \log n)$
- d) $O(n^2)$

Answer: b

Q6. Which method is used to sort a stack using only another stack?

- a) Push-Pop-Sort
- b) Insertion Sort using Stack
- c) Stack Sort Algorithm
- d) Recursive Sort

Answer: b

Q7. Which of these is required to implement multiple stacks in a single array efficiently?

- a) Dynamic Partitioning
- b) Fixed Partitioning
- c) Linked List
- d) Hashing

Answer: a

Q8. Which stack-based algorithm is used in strongly connected components detection?

- a) Tarjan's Algorithm
- b) Prim's Algorithm
- c) Dijkstra's Algorithm
- d) Bellman-Ford Algorithm

Answer: a

Q9. Which stack-based algorithm detects articulation points?

- a) DFS with low-link values
- b) BFS
- c) Kruskal's Algorithm
- d) Floyd-Warshall

Answer: a

Q10. What is the space complexity of evaluating a postfix expression with n operands and operators?

- a) $O(1)$
- b) $O(\log n)$
- c) $O(n)$
- d) $O(n^2)$

Answer: c

Q11. Which of these operations is costly in stack implemented using singly linked list?

- a) Push
- b) Pop
- c) Traversal
- d) Peek

Answer: c

Q12. Which stack structure is used in non-recursive tree traversal?

- a) Explicit Stack
- b) Call Stack

- c) Operand Stack
- d) Heap

Answer: a

Q13. Which is the worst-case time complexity of push and pop in amortized dynamic array stack?

- a) $O(1)$
- b) $O(n)$
- c) $O(\log n)$
- d) $O(n \log n)$

Answer: a

Q14. Which stack algorithm is used in compiler syntax parsing?

- a) Shift-Reduce Parsing
- b) BFS Parsing
- c) DFS Parsing
- d) Backtracking Parsing

Answer: a

Q15. Which stack-based algorithm is used for evaluating Boolean expressions?

- a) Shunting Yard
- b) Two-Stack Boolean Evaluation
- c) Recursive Descent
- d) None

Answer: b

Q16. Which is the maximum stack depth for a recursive factorial function with input n ?

- a) $O(1)$
- b) $O(n)$
- c) $O(\log n)$
- d) $O(n^2)$

Answer: b

Q17. Which of these expressions requires maximum stack depth during evaluation?

- a) Fully parenthesized left-associative
- b) Fully parenthesized right-associative
- c) Balanced tree structure
- d) None

Answer: b

Q18. Which method can prevent stack overflow in recursion?

- a) Iteration
- b) Tail Recursion Optimization
- c) Increasing stack size
- d) All of the above

Answer: d

Q19. Which stack-based algorithm is used in N-Queens problem?

- a) Backtracking using recursion stack
- b) BFS
- c) Kruskal's
- d) Dijkstra's

Answer: a

Q20. Which stack-based algorithm is used to evaluate expression trees?

- a) Postorder traversal with stack
- b) Preorder traversal with stack
- c) Inorder traversal
- d) Level-order traversal

Answer: a

Q21. Which postfix expression evaluation requires maximum stack size for n operands?

- a) Balanced
- b) Skewed
- c) Sequential operators
- d) Sequential operands

Answer: d

Q22. Which stack operation is most critical in parsing algorithms?

- a) Push
- b) Pop
- c) Peek
- d) Both A and B

Answer: d

Q23. Which algorithm uses stacks in its non-recursive form?

- a) DFS
- b) BFS
- c) Prim's
- d) Kruskal's

Answer: a

Q24. Which stack is used internally by virtual machines like JVM?

- a) Operand Stack
- b) Call Stack
- c) System Stack
- d) Data Stack

Answer: a

Q25. Which stack application requires $O(n^2)$ in worst case?

- a) Sorting a stack using another stack
- b) Expression evaluation
- c) Recursion
- d) DFS

Answer: a

Q26. Which is an advanced application of stacks?

- a) Detecting strongly connected components
- b) Finding articulation points
- c) Backtracking algorithms
- d) All of the above

Answer: d

Q27. Which stack is used in divide-and-conquer algorithms?

- a) Recursion Stack
- b) Operand Stack
- c) Explicit Stack
- d) System Stack

Answer: a

Q28. Which is postfix for $A+(B*(C^D-E))$?

- a) $ABCD^*E-+$
- b) $A+BCD^*-E$
- c) $AB+C^*D^*-E$
- d) $ABCD^*^*E-+$

Answer: a

Q29. Which stack-based algorithm helps in checking XML/HTML tags?

- a) Balanced Parentheses Check
- b) Tag Matching Stack
- c) Parsing Stack

d) Expression Tree Stack

Answer: b

Q30. Which problem cannot be solved by a single stack?

a) Palindrome check

b) Balanced parentheses

c) Expression evaluation

d) BFS traversal

Answer: d

Q31. Which prefix represents $(A+B)*(C-D)$?

a) $*+AB-CD$

b) $+A*BCD$

c) $*AB+CD-$

d) $AB+CD-*$

Answer: a

Q32. Which stack-based algorithm is used for iterative quicksort?

a) Explicit Stack for Subranges

b) Recursive Stack

c) Operand Stack

d) None

Answer: a

Q33. Which stack algorithm detects next greater element in array?

a) Monotonic Stack

b) Balanced Parentheses Stack

c) Expression Evaluation Stack

d) Heap Stack

Answer: a

Q34. Which stack algorithm is used in Largest Rectangle in Histogram problem?

a) Monotonic Stack

b) Recursive Stack

c) Queue Stack

d) None

Answer: a

Q35. Which stack-based algorithm is used in Stock Span Problem?

a) Monotonic Decreasing Stack

- b) Recursive Stack
- c) Heap-based stack
- d) None

Answer: a

Q36. Which of these requires multiple stacks for efficient solution?

- a) Undo-Redo operations
- b) DFS
- c) Expression Evaluation
- d) Tower of Hanoi

Answer: a

Q37. Which stack-based method is used for strongly connected components?

- a) Tarjan's Algorithm
- b) Kosaraju's Algorithm
- c) Both
- d) None

Answer: c

Q38. Which is the time complexity of Tower of Hanoi recursion stack usage?

- a) $O(2^n)$
- b) $O(n)$
- c) $O(\log n)$
- d) $O(n^2)$

Answer: a

Q39. Which stack problem requires maintaining min element in $O(1)$?

- a) Special Stack (Min Stack)
- b) Regular Stack
- c) Queue
- d) Max Heap

Answer: a

Q40. Which stack algorithm finds nearest smaller to left in array?

- a) Monotonic Stack
- b) DFS
- c) BFS
- d) Binary Search

Answer: a

Q41. Which stack algorithm is used to check duplicate parentheses?

- a) Parentheses Matching with Stack
- b) Expression Evaluation
- c) Queue Matching
- d) None

Answer: a

Q42. Which stack-based algorithm finds maximum area under histogram?

- a) Monotonic Stack
- b) Heap Stack
- c) DFS Stack
- d) None

Answer: a

Q43. Which stack optimization avoids memory overflow?

- a) Tail Recursion Elimination
- b) Linked Stack
- c) Garbage Collection
- d) Paging

Answer: a

Q44. Which postfix requires max stack size: $ABC*+DE/-$?

- a) 2
- b) 3
- c) 4
- d) 5

Answer: b

Q45. Which stack is used in parsing mathematical expressions in compilers?

- a) Operator Stack
- b) Operand Stack
- c) Both
- d) None

Answer: c

Q46. Which problem requires stack + dynamic programming hybrid?

- a) Longest Valid Parentheses
- b) Expression Evaluation
- c) DFS
- d) BFS

Answer: a

Q47. Which stack-based algorithm is efficient in matching brackets of different types?

- a) Balanced Parentheses Stack
- b) Expression Parsing Stack
- c) Symbol Matching Stack
- d) None

Answer: a

Q48. Which stack is used in context switching in OS?

- a) Call Stack
- b) Kernel Stack
- c) User Stack
- d) Both B and C

Answer: d

Q49. Which stack algorithm uses lazy propagation for operations?

- a) Segment Tree with Stack
- b) Monotonic Stack
- c) Tarjan's Algorithm
- d) None

Answer: a

Q50. Which stack-based problem has $O(n)$ solution with $O(n)$ extra space?

- a) Next Greater Element
- b) Largest Rectangle in Histogram
- c) Stock Span
- d) All of the above

Answer: d