

MCQs on stacks with answers

1. The postfix form of the expression $(A + B) * (C * D - E) * F / G$ is?

- a) $AB + CD * E - FG / **$
- b) $AB + CD * E - F ** G /$
- c) $AB + CD * E - * F * G /$
- d) $AB + CDE * - * F * G /$

[View Answer](#) / [Hide Answer](#)

ANSWER: a) $AB + CD * E - FG / **$

2. The data structure required to check whether an expression contains balanced parenthesis is?

- a) Stack
- b) Queue
- c) Array
- d) Tree

[View Answer](#) / [Hide Answer](#)

ANSWER: a) Stack

3. What data structure would you mostly likely see in a non recursive implementation of a recursive algorithm?

- a) Linked List
- b) Stack
- c) Queue
- d) Tree

[View Answer](#) / [Hide Answer](#)

ANSWER: b) Stack

4. The process of accessing data stored in a serial access memory is similar to manipulating data on a -----?

- a) Heap
- b) Binary Tree
- c) Array
- d) Stack

[View Answer](#) / [Hide Answer](#)

ANSWER: d) Stack

5. The postfix form of $A*B+C/D$ is?

- a) $*AB/CD+$
- b) $AB*CD/+$
- c) $A*BC+/D$
- d) $ABCD+/*$

[View Answer](#) / [Hide Answer](#)

ANSWER: b) $AB*CD/+$

6. Which data structure is needed to convert infix notation to postfix notation?

- a) Branch
- b) Tree
- c) Queue
- d) Stack

[View Answer](#) / [Hide Answer](#)

ANSWER: d) Stack

7. The prefix form of $A-B / (C * D \wedge E)$ is?

- a) $-/*\wedge ACBDE$
- b) $-ABCD*\wedge DE$
- c) $-A/B*C\wedge DE$
- d) $-A/BC*\wedge DE$

[View Answer](#) / [Hide Answer](#)

ANSWER: c) $-A/B*C\wedge DE$

8. What is the result of the following operation

Top (Push (S, X))

- a) X
- b) Null
- c) S
- d) None

[View Answer](#) / [Hide Answer](#)

ANSWER: a) X

9. The prefix form of an infix expression $p + q - r * t$ is?

- a) $+pq - *rt$
- b) $- +pqr * t$
- c) $- +pq * rt$
- d) $- + * pqrt$

[View Answer](#) / [Hide Answer](#)

ANSWER: c) $- +pq * rt$

10. Which data structure is used for implementing recursion?

- a) Queue
- b) Stack
- c) Array
- d) List

[View Answer](#) / [Hide Answer](#)

ANSWER: b) Stack

11. The result of evaluating the postfix expression 5, 4, 6, +, *, 4, 9, 3, /, +, * is?

- a) 600
- b) 350
- c) 650
- d) 588

[View Answer](#) / [Hide Answer](#)

ANSWER: b) 350

12. Convert the following infix expressions into its equivalent postfix expressions

$(A + B \wedge D) / (E - F) + G$

- a) $(A B D \wedge + E F - / G +)$
- b) $(A B D + \wedge E F - / G +)$
- c) $(A B D \wedge + E F / - G +)$
- d) None

[View Answer](#) / [Hide Answer](#)

ANSWER: a) $(A B D \wedge + E F - / G +)$

13. Convert the following Infix expression to Postfix form using a stack

$x + y * z + (p * q + r) * s$, Follow usual precedence rule and assume that the expression is legal.

- a) $xyz^*+pq^*r+s^*+$
- b) $xyz^*+pq^*r+s+^*$
- c) $xyz+^*pq^*r+s^*+$
- d) none

[View Answer / Hide Answer](#)

ANSWER: a) $xyz^*+pq^*r+s^*+$

14. Which of the following statement(s) about stack data structure is/are NOT correct?

- a) Stack data structure can be implemented using linked list
- b) New node can only be added at the top of the stack
- c) Stack is the FIFO data structure
- d) The last node at the bottom of the stack has a NULL link

[View Answer / Hide Answer](#)

ANSWER: c) Stack is the FIFO data structure

15. Consider the linked list implementation of a stack. Which of the following node is considered as Top of the stack?

- a) First node
- b) Last node
- c) Any node
- d) Middle node

[View Answer / Hide Answer](#)

ANSWER: a) First node

16. Consider the following operation performed on a stack of size 5.

Push(1);

Pop();

Push(2);

Push(3);

Pop();

Push(4);

Pop();

Pop();

Push(5);

After the completion of all operation, the no of element present on stack are

a) 1

b) 2

c) 3

d) 4

[View Answer / Hide Answer](#)

ANSWER: a) 1

17. Which of the following is not an inherent application of stack?

a) Reversing a string

b) Evaluation of postfix expression

c) Implementation of recursion

d) Job scheduling

[View Answer / Hide Answer](#)

ANSWER: d) Job scheduling

18. Which of the following operation take worst case linear time in the array implementation of stack?

- a) Push
- b) Pop
- c) IsEmpty
- d) None

[View Answer](#) / [Hide Answer](#)

ANSWER: d) None

19. The type of expression in which operator succeeds its operands is?

- a) Infix Expression
- b) pre fix Expression
- c) postfix Expression
- d) None

[View Answer](#) / [Hide Answer](#)

ANSWER: c) postfix Expression

20. Which of the following application generally use a stack?

- a) Parenthesis balancing program
- b) Syntax analyzer in compiler
- c) Keeping track of local variables at run time
- d) All of the above

[View Answer](#) / [Hide Answer](#)

ANSWER: d) All of the above

21. Consider the following array implementation of stack:

```
#define MAX 10
```

```
Struct STACK
```

```
{
```

```
Int arr [MAX];
```

```
Int top = -1;
```

```
}
```

If the array index starts with 0, the maximum value of top which does not cause stack overflow is?

- a) 8
- b) 9
- c) 10
- d) 11

[View Answer / Hide Answer](#)

ANSWER: a) 8

22. What is the minimum number of stacks of size n required to implement a queue of size n?

- a) One
- b) Two
- c) Three
- d) Four

[View Answer / Hide Answer](#)

ANSWER: b) Two

23. Assume that the operators $+$, $-$, \times are left associative and \wedge is right associative. The order of precedence (from highest to lowest) is \wedge , \times , $+$, $-$. The postfix expression corresponding to the infix expression $a + b \times c - d \wedge e \wedge f$ is

- a) $abc \times + def \wedge \wedge -$
- b) $abc \times + de \wedge f \wedge -$
- c) $ab + c \times d - e \wedge f \wedge$
- d) $- + a \times bc \wedge \wedge def$

[View Answer](#) / [Hide Answer](#)

ANSWER: a) $abc \times + def \wedge \wedge -$

24. If the elements "A", "B", "C" and "D" are placed in a stack and are deleted one at a time, in what order will they be removed?

- a) ABCD
- b) DCBA
- c) DCAB
- d) ABDC

[View Answer](#) / [Hide Answer](#)

ANSWER: b) DCBA

25. Consider the usual implementation of parentheses balancing program using stack. What is the maximum number of parentheses that will appear on stack at any instance of time during the analysis of $((()((()((())))?)$

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

d) 4

[View Answer](#) / [Hide Answer](#)

ANSWER: c) 3