

Short Answer (1 Word or 1 Line)

1. Convert the following infix expression to postfix notation: $(A+B)/(C-D)$
2. You have an array of n integers. What is the space complexity of storing this array in memory?
3. What will be the time complexity of the following code?

```
int i = 0;
while (i < n) {
    i = i + 1;
    i = i * 2;
}
```
4. You have a recursive function that calls itself n times. What is the space complexity of the function call stack?
5. List two applications of a Stack in Real Life.
6. List two applications of a Queue in Real Life.
7. Convert the following infix expression to prefix notation: $(A+B)/(C-D)$
8. Convert the following infix expression to postfix notation: $(A+B)*C$
9. What will be the time complexity of the following code?

```
for (int i = 1; i <= n; i *= 2) {
    // Some operation
}
```
10. You have a 2D array of size $m \times n$. What is the space complexity of storing this 2D array in memory?
11. You are implementing a stack data structure that can hold up to n elements. What is the space complexity of the stack?
12. Which principle does a stack follow: LIFO or FIFO?

Long Answer (40 - 100 Words)

1. Explain the concept of a Data Structure. Discuss its importance in programming and provide examples of different types of data structures.
2. Differentiate between Primitive Data Types and Non-Primitive Data Types. Illustrate your answer with examples and explain how each type is used in programming.
3. Write an Algorithm to solve the Tower of Hanoi problem using recursion.
4. Define a data structure and explain its role in organizing and managing data in software applications. Discuss the various operations that can be performed on data structures and provide examples.
5. Differentiate between Linear Data Structures and Non-Linear Data Structures. Illustrate your answer with examples and explain how each type is used in various applications .
6. Write an Algorithm to solve the Fibonacci series problem using recursion.
7. Write an Algorithm to print strings in reverse order using stack.
8. Explain the concept of a queue and its applications. Include a description of queue operations and their significance in programming.
9. Write an Algorithm to implement the following queue operations: enqueue(), dequeue(), front(), and isEmpty().
10. Explain the different types of array implementations and their uses. Discuss the advantages and disadvantages of each type in practical scenarios.
11. Write an Algorithm to perform insert, delete, and display operations using a simple queue.
12. Explain the concept of a stack and its applications. Include a description of stack operations and their significance in programming.
13. Write an Algorithm to perform the stack operations: push(), pop(), peek(), and update().

14. Describe the different types of queues and their uses. Discuss the advantages and disadvantages of each type of queue in practical scenarios.