**🔥 All Possible Questions on Recursion 🔥**

This list covers **theory, coding problems, and advanced recursion concepts** to help you master recursion.

**🔹 Basic Theory Questions**

1. What is recursion? Explain with an example.
2. How does recursion work internally in memory?
3. What is a base case in recursion? Why is it important?
4. What happens if recursion has no base case?
5. What is **direct** and **indirect** recursion? Give examples.
6. What is **tail recursion** and **non-tail recursion**?
7. How does recursion differ from iteration?
8. What are the advantages of recursion?
9. What are the disadvantages of recursion?
10. What is **stack overflow** in recursion? How can we prevent it?
11. What are some real-world applications of recursion?
12. What are the differences between recursion and loops?
13. What is **memoization** in recursion? How does it help?
14. What is the **time complexity** of recursion?
15. What is the **space complexity** of recursion?
16. What is **functional recursion**?
17. When should recursion be avoided?
18. Why is recursion slower than iteration?
19. What is **tree recursion**? Give an example.
20. What is the role of **recursion depth** in programming?

**🔹 Coding Questions - Easy Level**

1. Write a program to **print numbers from N to 1** using recursion.
2. Write a program to **print numbers from 1 to N** using recursion.
3. Write a recursive function to find the **sum of first N natural numbers**.
4. Write a recursive function to find the **factorial of a number**.
5. Write a recursive function to **calculate power (a^b)**.
6. Write a recursive function to check if a **string is palindrome**.
7. Write a recursive function to find the **GCD (Greatest Common Divisor)** of two numbers.
8. Write a recursive function to find the **LCM (Least Common Multiple)** of two numbers.
9. Write a recursive function to compute the **nth Fibonacci number**.
10. Write a recursive function to find the **sum of digits of a number**.

**🔹 Coding Questions - Intermediate Level**

1. Write a recursive function to **reverse a string**.
2. Write a recursive function to **reverse an array**.
3. Write a recursive function to check if an **array is sorted**.
4. Write a recursive function to find the **maximum element in an array**.
5. Write a recursive function to find the **minimum element in an array**.
6. Implement **binary search** using recursion.
7. Implement **merge sort** using recursion.
8. Implement **quick sort** using recursion.
9. Solve the **Tower of Hanoi** problem using recursion.
10. Implement a recursive function to **generate all subsets** of a given set.

**🔹 Coding Questions - Advanced Level**

1. Write a recursive function to solve **N-Queens Problem**.
2. Solve the **Rat in a Maze** problem using recursion.
3. Implement **Depth First Search (DFS)** using recursion.
4. Implement **Breadth First Search (BFS)** using recursion (via recursion stack).
5. Solve the **Knight's Tour Problem** using recursion.
6. Write a recursive function to **generate permutations** of a given string.
7. Write a recursive function to solve the **Josephus problem**.
8. Write a recursive function to generate **all valid parenthesis combinations** for N pairs.
9. Solve the **Word Break Problem** using recursion.
10. Implement **Trie Insert and Search** using recursion.

**🔹 Miscellaneous Recursion Questions**

1. How does recursion work in **tree traversal** (Preorder, Inorder, Postorder)?
2. How is recursion used in **graph traversal** (DFS, BFS)?
3. What is **backtracking**? How does recursion help in backtracking problems?
4. What is **dynamic programming**? How can recursion be optimized with DP?
5. What is **tail call optimization**?
6. Why is **memoization** useful in recursion?
7. How is recursion used in **functional programming**?
8. How does recursion handle **multi-branch problems**?
9. How can recursion be converted to iteration?
10. What is **divide and conquer**? How does recursion support it?

**🎯 Summary of Recursion Problems Covered**

✅ **Basic recursion problems**  
✅ **Mathematical recursion problems**  
✅ **Sorting and searching using recursion**  
✅ **Tree and graph traversal using recursion**  
✅ **Backtracking problems**  
✅ **Advanced recursive problems (N-Queens, DFS, Word Break, etc.)**