1) What is Recursion?

2) List and Explain types of Recursion

3) Explain the data structures used to perform recursion?

4)List the examples where recursion is used

5)Explain the difference between Recursion and Iteration, justify which to use when,Tail recursion?

Lab Questions :

1. Write a program to print a series of numbers with recursive Java methods

2. Write a program to sum a series of numbers with Java recursion

3. Write a program to calculate a factorial in Java with recursion

4. Write a program to print the Fibonacci series with Java and recursion

5.Write a program to implement a recursive Java palindrome checker

1. What is Recursion?

The process in which a function calls itself directly or indirectly is called recursion and the corresponding function is called a recursive function. Using recursive algorithm, certain problems can be solved quite easily. Examples of such problems are [Towers of Hanoi (TOH)](http://quiz.geeksforgeeks.org/c-program-for-tower-of-hanoi/), [Inorder/Preorder/Postorder Tree Traversals](https://www.geeksforgeeks.org/tree-traversals-inorder-preorder-and-postorder/), [DFS of Graph](https://www.geeksforgeeks.org/depth-first-traversal-for-a-graph/), etc.

1. List and Explain types of Recursion ?

Recursion are mainly of**two types** depending on whether **a function calls itself from within itself** or **more than one function call one another mutually.** The first one is called **direct recursion** and another one is called **indirect recursion.**

1. Explain the data structures used to perform recursion?

**Stack is the data structure used to perform recursion**. Explanation: Because of its LIFO (Last In First Out) property, it remembers its 'caller' so knows whom to return when the function has to return. Recursion makes use of the system stack for storing the return addresses of the function calls.

1. List the examples where recursion is used?

People often **sort stacks of documents** using a recursive method. For example, imagine you are sorting 100 documents with names on them. First place documents into piles by the first letter, then sort each pile. Looking up words in the dictionary is often performed by a binary-search-like technique, which is recursive.

1. Explain the difference between Recursion and Iteration, justify which to use when,Tail recursion?

|  |  |  |
| --- | --- | --- |
| **On the basis of** | **Recursion** | **Iteration** |
| ****Basic**** | Recursion is the process of calling a function itself within its own code. | In iteration, there is a repeated execution of the set of instructions. In Iteration, loops are used to execute the set of instructions repetitively until the condition is false. |
| ****Syntax**** | There is a termination condition is specified. | The format of iteration includes initialization, condition, and increment/decrement of a variable. |
| ****Termination**** | The termination condition is defined within the recursive function. | Here, the termination condition is defined in the definition of the loop. |
| ****Code size**** | The code size in recursion is smaller than the code size in iteration. | The code size in iteration is larger than the code size in recursion. |
| ****Infinite**** | If the recursive function does not meet to a termination condition, it leads to an infinite recursion. There is a chance of system crash in infinite recursion. | Iteration will be infinite, if the control condition of the iteration statement never becomes false. On infinite loop, it repeatedly used CPU cycles. |
| ****Applied**** | It is always applied to functions. | It is applied to loops. |
| ****Speed**** | It is slower than iteration. | It is faster than recursion. |
| ****Usage**** | Recursion is generally used where there is no issue of time complexity, and code size requires being small. | It is used when we have to balance the time complexity against a large code size. |
| ****Time complexity**** | It has high time complexity. | The time complexity in iteration is relatively lower. We can calculate its time complexity by finding the no. of cycles being repeated in a loop. |
| ****Stack**** | It has to update and maintain the stack. | There is no utilization of stack. |
| ****Memory**** | It uses more memory as compared to iteration. | It uses less memory as compared to recursion. |
| ****Overhead**** | There is an extensive overhead due to updating and maintaining the stack. | There is no overhead in iteration. |