**Assignment2**

Date : 20-04-2022

**1)What is Recursion?**

Recursion in java is a process in which a method calls itself continuously. A method in java that calls itself is called recursive method.

It makes the code compact but complex to understand.

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.

**2) List and Explain types of Recursion**

**1. Direct Recursion: a function calls itself from within itself**

**4 types: Tail Recursion, Head Recursion, Tree Recursion, Nested Recursion**

**2. Indirect Recursion: more than one function call one another mutually**

**3) 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. Every recursive function has its equivalent iterative (non-recursive) function. Even when such equivalent iterative procedures are written, an explicit stack is to be used.

**4)List the examples where recursion is used**

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

**6) Difference between Linear & Binary Search**

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Linear Search** | **Binary Search** |
| 1 | It searches through the array/list from the beginning to the end. | It finds the position of the searched element by finding the middle element of the array. |
| 2 | elements don't need to be arranged in sorted order. | The pre-condition for the binary search is that the elements must be arranged in a sorted order. |
| 3 | The linear search can be implemented on any linear data structure such as an array, linked list, etc. | The implementation of binary search is limited as it can be implemented only on those data structures that have two-way traversal. |
| 4 | It is based on the sequential approach. | It is based on the divide and conquer approach. |
| 5 | It is preferrable for the small-sized data sets. | It is preferrable for the large-size data sets. |
| 6 | It is less efficient in the case of large-size data sets. | It is more efficient in the case of large-size data sets. |
| 7 | In a linear search, the worst- case scenario for finding the element is O(n). | In a binary search, the worst-case scenario for finding the element is O(log2n). |
| 8 | In a linear search, the best-case scenario for finding the first element in the list is O(1). | In a binary search, the best-case scenario for finding the first element in the list is O(1). |
| 9 | It can be implemented on both a single and multidimensional array. | It can be implemented only on a multidimensional array. |