**Aim**

The aim of this practical is to implement an adaptive search algorithm that automatically performs linear search for unsorted ~~collections~~ and binary search for sorted collections.

**Theory**

The adaptive search algorithm combines the efficiency of binary search for sorted collections and the simplicity of linear search for unsorted collections. It determines the nature of the given collection by checking if it is sorted or unsorted. Based on the result, it applies the appropriate search algorithm. If the collection is sorted, it performs binary search; otherwise, it resorts to linear search.

**Procedure**

The procedure for implementing the adaptive search algorithm can be summarized as follows:

1. Start by defining the target element that you want to find in the collection.
2. Check if the collection is sorted:
   1. If the collection is sorted, apply the binary search algorithm.
   2. If the collection is unsorted, apply the linear search algorithm.
3. If the binary search algorithm is applied:
   1. Initialize the search range, which initially includes the entire sorted collection.
   2. Divide the search range in half and compare the target element with the middle element of the range.
   3. Adjust the search range based on the comparison, reducing it by half with each iteration until the target element is found or the search range is empty.
   4. If the target element is found, return the index of the element. If the search range is empty, conclude that the element is not present.
4. If the linear search algorithm is applied:
   1. Iterate through each element in the unsorted collection, comparing it to the target element.
   2. If a match is found, return the index of the element. If the entire collection is traversed without finding a match, conclude that the element is not present

## Practice

**Example :**  
  
**If the Array is unsorted it performs Linear Search :**  
  
Suppose that the input array is = **[10, 13, 9, 18, 5, 11]**  
We have to find the element **18** in the array.

**If the Array is sorted it performs Binary Search :**  
  
Suppose that the input array is = **[17, 32, 45, 53, 57, 61, 79, 88]**  
We have to find the element **78** in the array.

In Your turn: Mention linear/binary search used

**Result**

The adaptive search algorithm will either return the index of the target element if it is found in the collection or -1 if the element is not present

In Quiz:

Q5 When can binary search be applied?

 Only on unsorted lists

 Only on lists of integers

 Only on sorted lists

 Only on small lists

Q8

Q8) What are the advantages of Linear Search Over Binary Search?

~~The array is ordered~~

 Less number of comparison

 less time and space complexity

 Linear search can be used irrespective of whether the array is sorted or not

**Applications : is it there in rests of the vlab: Check template**

**Add team and tools used on icon bar and add text**

**References**

**~~Here are some references for further exploration~~:**

* Introduction to Algorithms (CLRS): by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
* Grokking Algorithms: by Aditya Bhargava
* The Algorithm Design Manual: by Steven S. Skiena

**Could you try to add dynamic simulations in the vlsb?**