

Linear Search in Java

Linear search is used to search a key element from multiple elements. Linear search is less used today because it is slower than binary search and hashing.

Algorithm:

- Step 1: Traverse the array
- Step 2: Match the key element with array element.
- Step 3: If key element is found, return the index position of the array element.
- Step 4: If key element is not found, return -1

```
public class LinearSearchExample{
    public static int linearSearch(int[] arr, int key){
        for(int i=0;i<arr.length;i++){
            if(arr[i] == key){
                return i;
            }
        }
        return -1;
    }
    public static void main(String a[]){
        int[] a1= {10,20,30,50,70,90};
        int key = 50;
        System.out.println(key+" is found at index: "+linearSearch(a1, key));
    }
}
```

Output:

```
50 is found at index: 3
```

Binary Search in Java

Binary search is used to search a key element from multiple elements. Binary search is faster than linear search.

In case of binary search, array elements must be in ascending order. If you have unsorted array, you can sort the array using *Arrays.sort(arr)* method.

```
class BinarySearchExample{  
    public static void binarySearch(int arr[], int first, int last, int key){  
        int mid = (first + last)/2;  
        while( first <= last ){  
            if ( arr[mid] < key ){  
                first = mid + 1;  
            }else if ( arr[mid] == key ){  
                System.out.println("Element is found at index: " + mid);  
                break;  
            }else{  
                last = mid - 1;  
            }  
            mid = (first + last)/2;  
        }  
        if ( first > last ){  
            System.out.println("Element is not found!");  
        }  
    }  
}
```

```
public static void main(String args[]){  
    int arr[] = {10,20,30,40,50};  
    int key = 30;  
    int last=arr.length-1;  
    binarySearch(arr,0,last,key);  
}  
}
```

Output:

```
Element is found at index: 2
```