Assignment-1

1.Sort elements using bubble sort

```
package Assignment4;
import java.util.*;
public class BubbleSort {

	public static void main(String[] args) {
		Scanner scan=new Scanner(System.in);
		System.out.println("How many elements u need");
		int n=scan.nextInt();
		System.out.println("Enter "+n+" Elements");
		int a[]=new int[n];
		for(int i=0;i<n;i++) {
			a[i]=scan.nextInt();
		}
		bubbleSort(a,n);

	}
	public static void bubbleSort(int a[],int n) {
		for(int i=0;i<n-1;i++) {
			for(int j=0;j<n-1-i;j++) {
				if(a[j]>a[j+1]) {
					int temp=a[j];
					a[j]=a[j+1];
					a[j+1]=temp;
				}
			}

		}
		System.out.println("The sorted elements are");
		for(int i=0;i<n;i++)
			System.out.print(a[i]+" ");
	}

}
```

Output

```
How many elements u need
```

```
5
Enter 5 Elements
5 4 3 2 1
The sorted elements are
1 2 3 4 5
```

2.Linear Search

```
package Assignment4;
import java.util.*;
public class LinearSearch {

	public static void main(String[] args) {
		System.out.println("Enter how many elements u
need");
		Scanner scan=new Scanner(System.in);
		int n=scan.nextInt();
		System.out.println("Enter "+n+" Elements");
		int a[]=new int[n];
		for(int i=0;i<n;i++) {
			a[i]=scan.nextInt();
		}
		System.out.println("Enter key to be searched");
		int key=scan.nextInt();
	   System.out.println(linearSearch(a,n,key));
	}
	public static boolean linearSearch(int a[],int n,int
key) {
		for(int i=0;i<n;i++) {
			if(key==a[i])
				return true;
		}
		return false;
	}

}
```

Output

```
Enter how many elements u need
6
Enter 6 Elements
1
2
3
4
5
```

```
6
Enter key to be searched
6
true
```

3.Binary search for integers and strings

```
package Assignment4;
import java.util.*;
public class BinarySearch {

     public static void main(String[] args) {
          Scanner scan=new Scanner(System.in);
          System.out.println("1.Binary Search for integers
2.Binary search for strings 3.exit");
          System.out.println("Enter choice");
          int n=scan.nextInt();
          switch(n) {
          case 1:System.out.println("Enter how many elements
u need");
                int ele=scan.nextInt();
                System.out.println("Enter elements");
                int a[]=new int[ele];
                for(int i=0;i<ele;i++) {
                 a[i]=scan.nextInt();
                }
                System.out.println("Enter key to be
searched");
                int key=scan.nextInt();
                System.out.println(findElement(a,key));
                break;
          case 2:System.out.println("Enter how many names u
need");
                 int n1=scan.nextInt();
                 scan.nextLine();
                 System.out.println("Enter names");
                 String names[]=new String[n1];
                 for(int i=0;i<n1;i++) {
                  names[i]=scan.nextLine();
                 }
                 System.out.print("Enter name to be
searched");
                 String name=scan.nextLine();
                 System.out.println(findString(names,name));
```

```
                break;
          case 3:break;


          }

     }
     public static boolean findElement(int a[],int key) {
        int low=0,high=a.length-1;
        boolean flag=false;
        int mid=(low+high)/2;
        while(low<=high) {
             if(a[mid]==key)
                  return true;
             else if(a[mid]<key) {
                  low=mid+1;
             }
             else {
                  high=mid-1;
             }
             mid=(low+high)/2;
        }
        return flag;
     }
    public static boolean findString(String names[],String
name) {
     int low=0,high=names.length-1;
        boolean flag=false;
        int mid=(low+high)/2;
        while(low<=high) {
             if(names[mid].equals(name))
                  return true;
             else if(names[mid].compareTo(name)<0) {
                  low=mid+1;
             }
             else {
                  high=mid-1;
             }
             mid=(low+high)/2;
```

```
        }
        return flag;
    }
}
```

Output

```
1.Binary Search for integers 2.Binary search for strings
3.exit
Enter choice
1
Enter how many elements u need
5
Enter elements
1
2
3
4
5
Enter key to be searched
5
true
1.Binary Search for integers 2.Binary search for strings
3.exit
Enter choice
2
Enter how many names u need
5
Enter names
bharath
chaaru
praveen
satish
sharath
Enter name to be searchedsatish
true
```

4.Insertion sort

```
package Assignment4;
import java.util.*;
public class InsertionSort {

	public static void main(String[] args) {
		System.out.println("Enter how many elements u
need");
		Scanner scan=new Scanner(System.in);
		int n=scan.nextInt();
		System.out.println("Enter "+n+" Elements");
		int a[]=new int[n];
		for(int i=0;i<n;i++) {
			a[i]=scan.nextInt();
		}
		insertionSort(a,n);

	}
	public static void insertionSort(int a[],int n) {
		int j,temp;
		for(int i=1;i<n;i++) {
			j=i-1;
			temp=a[i];
			while(j>=0 && a[j]>temp) {
				a[j+1]=a[j];
				j--;
			}
			a[j+1]=temp;
		}
		System.out.println("Sorted using insertion sort
are");
		for(int i=0;i<n;i++)
			System.out.print(a[i]+" ");
	}

}
```

Output

```
Enter how many elements u need
5
Enter 5 Elements
5
4
3
2
1
Sorted using insertion sort are
1 2 3 4 5
```