Longest Increasing Subsequence

```
package longestSubseq;
import java.util.*;

//class subseq to perform the longest sub sequence operation
class Subseq{
	//int return type method which will return the length
	int Lis(int arr[], int n)
	{
		//created an array arrseq to store the lenght of all subseq
		int arrseq[] = new int[arr.length];

		//filling the arrseq with value 1
		Arrays.fill(arrseq, 1);

		//taking max = -1 because no lenght will be -1
		int max = -1;

		//performed the operation checking from j to i
		for(int i=1; i<arr.length; i++)
		{
			for(int j=0; j<i; j++)
			{
				if(arr[j]<arr[i])
				{
					arrseq[i] = Math.max(arrseq[i], arrseq[j]+1);
				}
			}
			//after every iteration we are comparing the current lenght with prev length.
			max = Math.max(max,arrseq[i]);
		}
		//returning the LIS
		return max;
	}
```

```
}



public class LongIncSub {
	public static void main(String[] args)
	{

		Scanner sc = new Scanner(System.in);
		System.out.println("enter number of values you want to store. ");



		int n = sc.nextInt();
		int arr[] = new int[n];

		//taking all the input from the user in an array
		for(int i=0; i<n; i++)
		{
			arr[i] = sc.nextInt();
		}

		//making object of class Subseq
		Subseq obj = new Subseq();
		System.out.println("Your Longest Common Subsequence is: ");

		//calling the lis method in class Subseq
		int ans = obj.Lis(arr, arr.length);
		//printing the ans
		System.out.println(ans);



	}
}
```