SOURCE CODE FOR LONGEST INCREASING SUBSEQUENCE:

```
package Longest_Increasing_Subsequence;
import java.util.Scanner;

public class  Longest_IncreasingSubsequence
{
    public int[] lis(int[] X)
    {
        int n = X.length - 1;
        int[] M = new int[n + 1];
        int[] P = new int[n + 1];
        int L = 0;

        for (int i = 1; i < n + 1; i++)
        {
            int j = 0;

            for (int pos = L ; pos >= 1; pos--)
            {
                if (X[M[pos]] < X[i])
                {
                    j = pos;
                    break;
                }
            }
            P[i] = M[j];
            if (j == L || X[i] < X[M[j + 1]])
            {
                M[j + 1] = i;
                L = Math.max(L,j + 1);
            }
        }


        int[] result = new int[L];
        int pos = M[L];
        for (int i = L - 1; i >= 0; i--)
        {
            result[i] = X[pos];
            pos = P[pos];
        }
        return result;
    }

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Ok! Let's do the Longest Increasing Subsequence
Algorithm Test\n");

        System.out.println("Enter number of elements");
        int n = sc.nextInt();
        int[] arr = new int[n + 1];
        System.out.println("\nEnter "+ n +" elements");
        for (int i = 1; i <= n; i++)
            arr[i] = sc.nextInt();
```

```
        Longest_IncreasingSubsequence obj = new
Longest_IncreasingSubsequence();
        int[] result = obj.lis(arr);


        System.out.print("\nLongest Increasing Subsequence : ");
        for (int i = 0; i < result.length; i++)
            System.out.print(result[i] +" ");
        System.out.println();
    }
}
```