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

public class P1S3P1 {

  public static List<Integer> findLongestIncreasingSubsequence(int[] nums) {
        if (nums == null || nums.length == 0)
            return new ArrayList<>();

        int n = nums.length;
        List<List<Integer>> subsequences = new ArrayList<>();
        List<Integer> longestIncreasingSubsequence = new ArrayList<>();

        for (int i = 0; i < n; i++) {
            List<Integer> currentSubsequence = new ArrayList<>();
            currentSubsequence.add(nums[i]);

            for (int j = 0; j < i; j++) {
                if (nums[i] > nums[j] && subsequences.get(j).size() >=
currentSubsequence.size()) {
                    currentSubsequence = new ArrayList<>(subsequences.get(j));
                    currentSubsequence.add(nums[i]);
                }
            }

            subsequences.add(currentSubsequence);

            if (currentSubsequence.size() > longestIncreasingSubsequence.size()) {
                longestIncreasingSubsequence = currentSubsequence;
            }
        }

        return longestIncreasingSubsequence;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the number of elements in the list: ");
        int n = scanner.nextInt();

        int[] nums = new int[n];
        System.out.println("Enter the elements of the list:");
        for (int i = 0; i < n; i++) {
            nums[i] = scanner.nextInt();
        }

        List<Integer> lis = findLongestIncreasingSubsequence(nums);
        System.out.println("Longest Increasing Subsequence:");
        System.out.println(lis);

        scanner.close();
    }
}
```