|  |
| --- |
| using System;  using System.Linq;  namespace LongestIncreasingSubsequence  {  class Program  {  public static void Main()  {  int[] subsequence = Console.ReadLine()  .Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries)  .Select(x => int.Parse(x))  .ToArray();  Console.WriteLine(string.Join(' ', subsequence)); //remove  int[] longestIncreasingSubsequence;  int[] length = new int[subsequence.Length];  int[] previousElement = new int[subsequence.Length];  int maxLength = 0;  int lastIndex = -1;  for (int i = 0; i < subsequence.Length; i++)  {  length[i] = 1;  previousElement[i] = -1;  for (int j = 0; j < i; j++)  {  if (subsequence[j] < subsequence[i] && length[j] >= length[i])  {  length[i] = 1 + length[j];  previousElement[i] = j;  }  }  if (length[i] > maxLength)  {  maxLength = length[i];  lastIndex = i;  }  }  longestIncreasingSubsequence = new int[maxLength];  for (int i = 0; i < maxLength; i++)  {  longestIncreasingSubsequence[i] = subsequence[lastIndex];  lastIndex = previousElement[lastIndex];  }  Array.Reverse(longestIncreasingSubsequence);  Console.WriteLine(string.Join(" ", longestIncreasingSubsequence));  }  }  } |