

2011年计算机二级C 辅导实例编程 (27) PDF转换可能丢失  
图片或格式, 建议阅读原文

[https://www.100test.com/kao\\_ti2020/646/2021\\_2022\\_2011\\_E5\\_B9\\_B4\\_E8\\_AE\\_A1\\_c97\\_646741.htm](https://www.100test.com/kao_ti2020/646/2021_2022_2011_E5_B9_B4_E8_AE_A1_c97_646741.htm) 最长上升子序列LIS算法实现 最长上升子序列问题是各类信息学竞赛中的常见题型, 也常常用来做介绍动态规划算法的引例, 笔者接下来将会对POJ上出现过的这类题目做一个总结, 并介绍解决LIS问题的两个常用算法( $n^2$ )和( $n\log n$ ). 问题描述:给出一个序

列 $a_1, a_2, a_3, a_4, a_5, a_6, a_7, \dots, a_n$ , 求它的一个子序列(设为 $s_1, s_2, \dots, s_n$ ), 使得这个子序列满足这样的性质,  $s_1 < s_2 < \dots < s_n$  例如有一个序列:1 7 3 5 9 4 8, 它的最长上升子序列就是 1 3 4 8 长度为4. 算法1( $n^2$ ):我们依次遍历整个序列, 每一次求出从第一个数到当前这个数的最长上升子序列, 直至遍历到最后一个数字为止, 然后再取dp数组里最大的那个即为整个序列的最长上升子序列。我们用dp[i]来存放序列1-i的最长上升子序列的长度, 那

么 $dp[i] = \max(dp[j] + 1, (j \in [1, i-1]))$ . 显然 $dp[1] = 1$ , 我们从 $i = 2$ 开始遍历后面的元素即可。下面是模板: //最长上升子序列( $n^2$ )

模板 //入口参数: 1.数组名称 2.数组长度(注意从1号位置开始)

```
template<int LIS(T a[],int n) { int i,j. int ans=1. int m=0. int *dp=new int[n+1]. dp[1]=1. for(i=2;i<=n;i++) for(j=1;j<i;j++) if(a[j]<a[i]) dp[i]=max(dp[j]+1, dp[i]); ans=max(ans, dp[i]); return ans; }
```

100Test 下载频道开通, 各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)