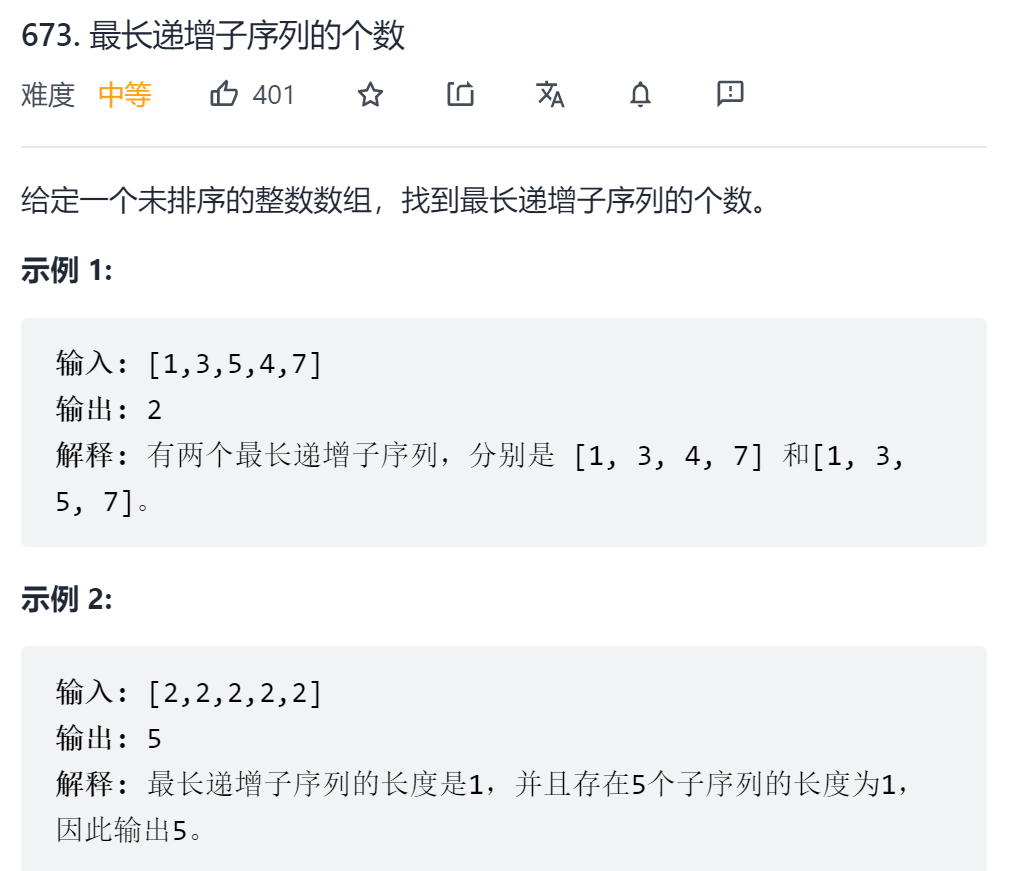
# 动态规划-两个一维动态数组

## 题目描述：



## 题目分析：

class Solution {  
public:  
 int findNumberOfLIS(vector<int>& nums) {  
 //动态规划  
 //两个动态数组  
 //dp[i]表示的是以nums[i]结尾的最长子序列的长度  
 //count[i]表示的是以nums[i]结尾的最长子序列长度为dp[i]的子序列的个数  
 int n=nums.size();  
 vector<int>dp(n,0),count(n,0);  
 int ans=0,maxlen=0;//maxlen记录最长子序列的长度  
 for(int i=0;i<n;i++){  
 dp[i]=1;//初始化  
 count[i]=1;  
  
 for(int j=0;j<i;j++){  
 if(nums[i]>nums[j]){  
 if(dp[j]+1>dp[i]){  
 dp[i]=dp[j]+1;  
 count[i]=count[j];  
 }else if(dp[j]+1==dp[i]){  
 count[i]+=count[j];  
 }  
 }  
 }  
  
 if(dp[i]>maxlen){  
 ans=count[i];  
 maxlen=dp[i];  
 }else if(dp[i]==maxlen){  
 ans+=count[i];  
 }  
   
 }  
 return ans;  
 }  
};