

C1-E题题解

班级: 222112

学号: 22373100

姓名: 郑悦

· 题目

套娃!

时间限制: 1000 ms 内存限制: 65536 kb

总通过人数: 148 总提交人数: 155

题目描述

我们知道俄罗斯套娃会把娃娃层层嵌套，外层的套娃肯定需要比内层的大。现在我们按顺序给出 n 个套娃，每个套娃的大小为 a_i ($1 \leq i \leq n$)，数据保证 a_i 互不相同。你要求出在套第 i 个套娃时，需要将其放在第几层（将最内层记为第 1 层）。

形式化的描述：对于每个 a_i ，你要求出在 a_1, a_2, \dots, a_i 中小于等于 a_i 的元素个数。

输入

第一行为一个正整数 T ($1 \leq T \leq 10$)，表示数据组数。

对于每组数据：

第一行为一个正整数 n ($1 \leq n \leq 10^3$)，表示套娃的个数。

第二行为 n 个正整数 a_i ($1 \leq a_i \leq 10^9$)，表示第 i 个套娃的大小。

输出

对于每组数据，输出一行 n 个整数，表示第 i 个套娃应该放在第几层。

注意点:

输出的是在套第*i*个套娃的时候，**这个套娃在目前应该放的层数**，而不是将所有输入的在最后再进行排序，输出最终各个套娃的层数。

思路:

考虑每来一个娃就套一次，找他的位置（层数）。类似插入排序，升序找位子，依次在之前排好序的娃列里面找到对应层数。（顺使用二分法提高排序效率。

· 代码

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
#include <ctype.h>
int a[100005]; //记录套娃的大小

int BInsertSort(int* arr, int size, int th)
{
    int left = 0;
    int right = th - 1;

    //查找插入位置
    while (left <= right)
    {
        int mid = (left + right) / 2;
        if (arr[th] < arr[mid])
        {
            right = mid - 1;
        }
        else
        {
            left = mid + 1;
        }
    }

    //后移数据并插入
    int temp = arr[th];
    for (right = th; right > left; right--)
    {
        arr[right] = arr[right-1];
    }
    arr[left] = temp;
    return left;
}

int main(){
    int t, n, i;
    scanf("%d", &t);
    while(t--){
        scanf("%d", &n);
        for(i = 0; i < n; i++){
            scanf("%d", &a[i]);
        }
    }
}
```

```
        printf("%d ",BInsertSort(a, n, i) + 1);  
    }  
    printf("\n");  
}  
return 0;  
}
```