

第五章 数的存储与组织

第56课 二分查找函数

《信息学奥赛一本通·编程启蒙 C++版》

【例 56.1】 和为给定数

【题目描述】

现给出若干个整数，询问其中是否有一对数的和等于给定的数。

【输入格式】

共三行：

第一行是整数 $n(0 < n \leq 100,000)$ ，表示有 n 个整数。

第二行是 n 个整数。整数的范围是在 0 到 2×10^8 之间。

第三行是一个整数 $m(0 \leq m \leq 2^{30})$ ，表示需要得到的和。

【输出格式】

若存在和为 m 的数对，输出两个整数，小的在前，大的在后，中间用单个空格隔开。若有多个数对满足条件，选择数对中较小的数更小的。若找不到符合要求的数对，输出一行 "No"。

【样例输入】

```
4  
2 5 1 4  
6
```

【样例输出】

```
1 5
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int a[100005];
4. int main(){
5.     int n,x;
6.     int *b;
7.     cin>>n;
8.     for(int i=0;i<n;i++)
9.         cin>>a[i];
10.    cin>>x;
11.    sort(a,a+n);
12.    for(int i=0;i<n;i++){
13.        b=lower_bound(a,a+n,x-a[i]);
14.        if(b-a==i&&*(b+1)!=*b) continue;
15.        if(*b==x-a[i]){
16.            cout<<a[i]<<" "<<x-a[i];
17.            return 0;
18.        }
19.    }
20.    cout<<"No";
21.    return 0;
22.}
```

【例 56.2】 饮料

【题目描述】

小 B 出门游玩，他现在走到了 n 家饮品店前，第 i 家买饮料的花费为 cost_i ，他只能去花费不超过自己所带钱数额的店，但他现在不确定自己兜里有多少钱，所以现在给出 m 次询问，请问如果他有 a_i 元，他有几家饮品店可选？

【输入格式】

第一行两个整数分别表示 n, m

第二行 n 个整数，第 i 个表示 cost_i ，接下来 m 行，每行一个整数表示询问。 $1 \leq n, m, \text{cost}_i \leq 10^6, 1 \leq a_i \leq 10^9$

【输出格式】

对于每次询问回答小 B 有几家饮品店可选。

【样例输入】

```
5 2
1 2 3 4 5
3
5
```

【样例输出】

```
3
5
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int x,n,m,a[1000005];
4. int main() {
5.     scanf("%d %d",&n,&m);
6.     for (int i = 0; i < n; i++) scanf("%d", &a[i]);
7.     sort(a,a + n);
8.     for (int i = 0; i < m; i++){
9.         scanf("%d",&x);
10.        printf( "%d\n", upper_bound(a, a + n, x)-a);}
11.     return 0;
12.}
```

练 56.1 猜猜乐(趣味编程)

【题目描述】

格莱尔和尼克玩猜数游戏（1 ~ 100 之间的整数）。格莱尔先选了一个数如 56 写在纸上，尼克用二分法去猜。

请你编一程序，输入要猜的数，让计算机输出尼克使用二分法猜数的过程。

【输入格式】

一行一个整数，为 n 。若 n 不符合要求，则重新输入。

【输出格式】

输出若干行，每行一个整数，为此时二分得到的数，当二分得到的数为 n 时，输出“成功！”，否则当二分结束时，输出“失败！”。

【样例输入】

56

【样例输出】

50

75

62

56

成功!

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3.
4. int search(int b[],int len,int key)
5. {
6.     int high,low;
7.     high=len-1;
8.     low=0;
9.     int mid=len/2;
10.    while(high>=low)
11.    {
12.        mid=(high+low)/2; // 求出二分中间的数
13.        cout<<b[mid]<<endl; // 输出该数
14.        if(b[mid]==key) return mid;
15.        else if(b[mid]>key) // 如果该数大于所求的数
16.            high=mid-1; // 范围缩小
17.        else // 反之亦然
18.            low=mid+1;
19.    }
20.    return 0;
21.}
22.
23.main()
24.{
25.    const int MAX=100;
26.    int a[MAX],n;
27.    for(int i=0;i<MAX;i++) // 预处理
28.        a[i]=i+1;
29.    //cout<<"n=";
30.    do // 输入符合条件的n
31.    {
32.        cin>>n;
33.    }while(n<1||n>100);
34.    if(search(a,MAX,n))
35.        cout<<"成功! ";
36.    else
37.        cout<<"失败! ";
38.    return 0;
39.}
```

练 56.2 查找最接近的元素

【题目描述】

在一个非降序列中，查找与给定值最接近的元素。

【输入格式】

第一行包含一个整数 n ，为非降序列长度。 $1 \leq n \leq 100000$ 。

第二行包含 n 个整数，为非降序列各元素。所有元素的大小均在 $0 \sim 1,000,000,000$ 之间。第三行包含一个整数 m ，为要询问的给定值个数。 $1 \leq m \leq 10000$ 。

接下来 m 行，每行一个整数，为要询问最接近元素的给定值。所有给定值的大小均在 $0 \sim 1,000,000,000$ 之间。

【输出格式】

m 行，每行一个整数，为最接近相应给定值的元素值，保持输入顺序。若有多个值满足条件，输出最小的一个。

【样例输入】

```
3
2 5 8
2
10
5
```

【样例输出】

```
8
5
```

【代码实现】

```
1. #include <cstdio>
2. #include <algorithm>
3. using namespace std;
4. const int maxn = 1e5 + 10;
5. int n, m, b;
6. int a[maxn];
7. int main()
8. {
9.     scanf("%d", &n);
10.    for (int i = 0; i < n; ++i)
11.    {
12.        scanf("%d", &a[i]);
13.    }
14.    scanf("%d", &m);
15.    for (int i = 0; i < m; ++i)
16.    {
17.        scanf("%d", &b);
18.        if (b > a[n - 1]) {
19.            printf("%d\n", a[n - 1]);
20.            continue;
21.        }
22.        if (b <= a[0]) {
23.            printf("%d\n", a[0]);
24.            continue;
25.        }
26.        int x = lower_bound(a, a + n, b) - a;
27.        if (a[x] - b < b - a[x - 1]) {
28.            printf("%d\n", a[x]);
29.        }
30.        else {
31.            printf("%d\n", a[x - 1]);
32.        }
33.    }
34.    return 0;
35.}
```

谢谢!

—