

1244: 和为给定数

题目描述

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

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108之间。

第三行是一个整数 m ($0 \leq m \leq 230$)，表示需要得到的和。

输出

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。

输入样例

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

输出样例

```
1 5
```

解析

本题是二分的一个变体，包含的信息非常多，需要我们仔细分析。

首先观察输入样例，输入的数据是乱序的，而我们一般使用二分法时，需要数据是严格的升序，所以，第一步需要先将数据进行升序。此外，考虑到输出也是要求从小到大，因此我们是很必要进行排序的。

接下来寻找核心要素：

1、左边界：索引0

2、右边界：索引 $n-1$

3、搜索逻辑：我们在数组中寻找两个数字，他们的和正好等于目标值。这与一般的搜索不同，我们不再需要mid的值，而是需要 $right+left$ ，然后根据这个结果进行边界的移动。

如果结果大了，则缩小右边界。如果结果小了，则增加右边界。

编码

```
#include<bits/stdc++.h>

#define N 100001
```

```

using namespace std;
int a[N];
long long n, m;

bool Search(int left, int right) {
    left = 1;
    while (left < right) {
        int cur = a[left] + a[right];
        //直接找到目标
        if (cur == m) {
            cout << a[left] << " " << a[right] << endl;
            return true;
        }
        //和小于目标值, 增加左边界
        else if (cur < m) {
            left++;
        }
        //否则, 减少右边界
        else if (cur > m) {
            right--;
        }
    }
    return false;
}

int main() {

    cin >> n;
    for (int i = 1; i <= n; i++)
        cin >> a[i];
    cin >> m;
    //将所有数据从小到大进行排序
    sort(a + 1, a + 1 + n);
    //执行二分搜索
    if (!Search(1, n)) {
        //没有找到目标
        cout << "No" << endl;
    }

    return 0;
}

```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

