

## 事件推送

### 题目描述：

同一个数轴  $X$  上有两个点的集合  $A=\{A_1, A_2, \dots, A_n\}$  和  $B=\{B_1, B_2, \dots, B_m\}$ ,  $A_i$  和  $B_j$  均为正整数,  $A$ 、 $B$  已经按照从小到大排好序,  $A$ 、 $B$  均不为空, 给定一个距离  $R$ (正整数), 列出同时满足如下条件的所有  $(A_i, B_j)$  数对:

- 1)  $A_i \leq B_j$
- 2)  $A_i, B_j$  之间的距离小于等于  $R$
- 3) 在满足 1) 2) 的情况下, 每个  $A_i$  只需输出距离最近的  $B_j$
- 4) 输出结果按  $A_i$  从小到大的顺序排序

### 输入描述：

第一行三个正整数  $n, m, R$

第二行  $n$  个正整数, 表示集合  $A$

第三行  $m$  个正整数, 表示集合  $B$

### 输入限制：

$1 \leq R \leq 100000$ ,  $1 \leq n, m \leq 100000$ ,  $1 \leq A_i, B_j \leq 1000000000$

### 输出描述：

每组数对输出一行  $A_i$  和  $B_j$ , 以空格隔开

### 补充说明：

#### 示例 1

##### 输入：

```
4 5 5
1 5 5 10
1 3 8 8 20
```

##### 输出：

```
1 1
5 8
5 8
```

##### 说明：

```
#include<iostream>
#include<vector>
using namespace std;
int erfen(int a, vector<int>b) {
    int left{ 0 };
    int right = b.size();
    while (left < right) {
        int mid = (left + right) / 2;
        if (b[mid] >= a) {
```

```

        right = mid;
    }
    else {
        left = mid + 1;
    }
}
return left;
}

int main() {
    int n;
    int m;
    int r;
    cin >> n;
    cin >> m;
    cin >> r;
    vector<int> a(n);
    vector<int> b(m);
    for (int i{ 0 }; i < n; i++) {
        cin >> a[i];
    }
    for (int i{ 0 }; i < m; i++) {
        cin >> b[i];
    }
    for (int i{ 0 }; i < n; i++) {
        int temp = erfen(a[i], b);
        if (temp < m) {
            if (b[temp] - a[i] <= r) {
                cout << a[i] << ' ' << b[temp]<<'\n';
            }
        }
    }
}
}

```