

题目描述：

现有两组服务器 A 和 B ，每组有多个算力不同的 CPU ，其中 $A[i]$ 是 A 组第 i 个 CPU 的运算能力， $B[i]$ 是 B 组第 i 个 CPU 的运算能力。一组服务器的总算力是各 CPU 的算力之和。

为了让两组服务器的算力相等，允许从每组各选出一个 CPU 进行一次交换，求两组服务器中，用于交换的 CPU 的算力，并且要求从 A 组服务器中选出的 CPU ，算力尽可能小。

输入描述：

第一行输入为 $L1$ 和 $L2$ ，以空格分隔， $L1$ 表示 A 组服务器中的 CPU 数量， $L2$ 表示 B 组服务器中的 CPU 数量。

第二行输入为 A 组服务器中各个 CPU 的算力值，以空格分隔。

第三行输入为 B 组服务器中各个 CPU 的算力值，以空格分隔。

$1 \leq L1 \leq 10000$

$1 \leq L2 \leq 10000$

$1 \leq A[i] \leq 100000$

$1 \leq B[i] \leq 100000$

输出描述：

对于每组测试数据，输出两个整数，以空格分隔，依次表示 A 组选出的 CPU 算力、 B 组选出的 CPU 算力。

要求从 A 组选出的 CPU 的算力尽可能小。

补充说明：

保证两组服务器的初始总算力不同。
答案肯定存在。

示例 1

输入：

```
2 2
1 1
2 2
```

输出：

```
1 2
```

说明：

从 *A* 组中选出算力为 *1* 的 *CPU*，与 *B* 组中算力为 *2* 的进行交换，使两组服务器的算力都等于 *3*。

示例 2

输入：

```
2 2
1 2
2 3
```

输出：

```
1 2
```

说明：

示例 3

输入：

1 2

2

1 3

输出：

2 3

说明：

示例 4

输入：

3 2

1 2 5

2 4

输出：

5 4

import sys

m, n = map(int, input().strip().split())

a = list(map(int, input().strip().split()))

b = list(map(int, input().strip().split()))

sum_a = sum(a)

sum_b = sum(b)

total = (sum_a + sum_b) // 2

diff_a = sum_a - total

a.sort()

for ai in a:

 if ai - diff_a in b:

 print(ai, ai - diff_a)

 break