

# CSP2022模拟赛

## 一. 题目概况

中文题目名称	背包	移动细胞	异或和	序列
可执行文件名	bag.cpp	cell.cpp	xor.cpp	sequence.cpp
输入文件名	bag.in	cell.in	xor.in	sequence.in
输出文件名	bag.out	cell.out	xor.out	sequence.out
时间限制	1000ms	1000ms	1000ms	1000ms
空间限制	512M	512M	512M	512M

## 二. 注意事项:

- 1、文件名（程序名和输入输出文件名）必须使用英文小写。
- 2、C/C++中函数main()的返回值类型必须是int，程序正常结束时的返回值必须是 0。
- 3、结果比较方式：全文比较（过滤行末空格及文末回车）。
- 4、本次测试需要使用文件读写。

# 1 背包 (bag)

## 题目描述

有  $M$  个背包容量分别为  $X_i$  千克,  $N$  个物品的重量分别为  $Y_i$  千克, 请问最少需要多少个背包可以把  $N$  个物品全部装进去呢? 如果装不下请输出  $-1$ 。

## 输入格式

第一行两个数,  $N, M$  表示物品数量以及背包数量;

第二行  $N$  个数, 表示物品的重量;

第三行  $N$  个数, 表示背包的容量。

## 输出格式

一行一个整数, 表示最少需要的背包个数, 或者无解输出  $-1$ 。

## 输入 #1

```
3 3
7 3 9
5 10 9
```

## 输出 #1

```
2
```

## 输入 #2

```
1 2
10
5 5
```

## 输出 #2

```
-1
```

## 数据范围

对于 50% 的数据, 有  $1 \leq N \leq 10, 1 \leq M \leq 10$

对于 100% 的数据, 有  $1 \leq N \leq 20, 1 \leq M \leq 50, 1 \leq X_i, Y_i \leq 10^8$

## 2 移动细胞 (cell)

### 题目描述

有一个  $n \times m$  的矩阵，每个格子有一个细胞。每一列都有一段连续的细胞被染成黑色，而其他的细胞被染成了白色。Moon 认为这个矩阵是“美丽”的，当前仅当任何两个黑色的细胞可以通过黑色的细胞四联通互相到达。

但是有可能这个矩阵并不是“美丽”的，所以 Moon 需要进行一些操作使得矩阵变成“美丽”的。每一步操作中，Moon 可以选择把某一列的黑色细胞同时向上或者向下移动一格，但是不能把黑色细胞移动到矩阵外面。Moon 想知道最少需要多少次操作，可以使得矩阵变成“美丽”的。

### 输入格式

第一行包含两个整数， $n, m$  表示矩阵的行数和列数；

接下来  $m$  行，第  $i$  行两个整数  $s_i, t_i$  表示每一列黑色细胞的上下边界。

### 输出格式

一个整数，表示最少的操作数。

### 输入 #1

```
9 3
1 2
4 5
7 9
```

### 输出 #1

```
4
```

### 数据范围

对于 30% 的数据， $1 \leq n, m \leq 200$

对于 100% 的数据， $1 \leq n \times m \leq 10^6, 1 \leq n, m \leq 10^5, 1 \leq s_i, t_i \leq n$

## 3 异或和 (xor)

### 题目描述

给定两个序列  $A, B$ , 每个序列长度为  $n$ , 定义  $X(A)$  为  $A$  序列中所有数的异或和,  $X(B)$  为  $B$  序列中所有数的异或和。

现在对于  $1 \leq i \leq n$ , Moon 可以交换  $A_i$  和  $B_i$ , 请问  $X(A) + X(B)$  最大值是多少呢?

### 输入格式

第一行,  $n$ , 表示  $A, B$  序列的长度;

第2行,  $n$  个整数, 表示  $A_i$ ;

第3行,  $n$  个整数, 表示  $B_i$ ;

### 输出格式

一行, 一个整数, 表示  $X(A) + X(B)$  最大值。

### 输入 #1

```
2
1 1
2 2
```

### 输出 #1

```
6
```

### 数据范围

对于 30% 的数据,  $1 \leq n \leq 10$

对于 100% 的数据,  $1 \leq n \leq 10^5, 0 \leq A_i \leq 10^{18}$

## 4 序列 (sequence)

### 题目描述

你有一个长度为  $n$  的序列  $\{a_i\}$ ，你要从中选取一个单调不升的子序列（可以为空），使其价值最大。

一个序列的价值定义如下：选取第  $i$  个数会减少  $b_i$  的价值，但会获得  $c_{b_i}$  的价值。如果选取的子序列中有两个相同的数  $i$ ，那它们会合并成一个数  $i + 1$ ，并额外获得  $c_{i+1}$  的价值。它们会不断合并直到这个子序列的数两两不同。一个序列的价值即为选取过程中的价值和。

你不要求出这个子序列具体是什么，但是要求出这个价值的最大值。

### 输入格式

第一行两个数  $n, m$ ，表示序列长度和  $a_i$  的上限。

第二行  $n$  个数  $a_1, a_2, \dots, a_n$ 。

第三行  $n$  个数  $b_1, b_2, \dots, b_n$ 。

第四行  $n + m$  个数  $c_1, c_2, \dots, c_{n+m}$ 。

### 输出格式

一行一个整数，表示子序列的最大价值。

### 输入 #1

```
5 4
4 3 1 2 1
1 2 1 2 1
1 2 3 4 5 6 7 8 9
```

### 输出 #1

```
6
```

### 输入 #2

```
2 2
1 2
0 0
2 1 100 -100
```

### 输出 #2

```
2
```

## 输入 #3

```
5 4
4 3 2 1 1
0 2 6 7 4
12 12 12 6 -3 -5 3 10 -4
```

## 输出 #3

```
62
```

## 数据范围

对于 100% 的数据,  $1 \leq n, m \leq 2000, 1 \leq a_i \leq m, 0 \leq b_i \leq 5000, |c_i| \leq 5000$

- *Subtask 1* [15%]:  $1 \leq n, m \leq 20$
- *Subtask 2* [20%]:  $1 \leq n, m \leq 200$
- *Subtask 3* [5%]:  $a_i$  单调不降
- *Subtask 4* [20%]:  $b_i = 0$
- *Subtask 5* [40%]: 无特殊限制