

# A $a^b$ problem Ver.⑨

## 题目描述

今天是期中考试，Cirno 遇到了一个经典问题：给定三个整数  $a, b, p$ ，求  $a^b \bmod p$ 。

## 输入

一行三个整数  $a, b, p$ 。

## 输出

一个整数，表示答案。

## 样例

### 输入

```
7 6 12
```

### 输出

```
1
```

## 数据范围

$0 \leq a, b < 2^{31}$ ,  $a + b > 0$ ,  $2 \leq p < 2^{31}$ 。

## HINT

如果你 TLE 了，不妨试试下面这个函数，它可以在  $O(\log n)$  的时间内计算模幂。

```
#include <stdio.h>

long long qpow(long long a, long long b, long long p)
{
    long long ans = 1;
    a = a % p;
    while (b)
    {
        if (b & 1)
            ans = (ans * a) % p;
        b >>= 1;
        a = a * a % p;
    }
}
```

```
    return ans;
}

int main()
{
    //补充你的代码
    return 0;
}
```

如果你很闲想看看实现[原理](#)。

## B *ljh*要复习

### 题目描述

老家伙*ljh*做了一个梦，在梦里他还剩 $m$ 个小时就要参加一门期末考试了，可他现在什么都不会，于是他从万能的打印店里搞到了一份总计 $n$ 页的复习资料，并制定了详细的复习计划：在 $1 \dots m$ 个小时中的第 $i$ 个小时，他会看完复习资料的 $L_i$ 到 $R_i$ 页。现在*ljh*想知道， $m$ 个小时后自己能不能看完复习资料，以及自己看的最多的是哪一页，设其为 $x$ （显然 $x$ 可能存在多个，*ljh*只想知道最小的那个）。

### 输入

第一行，两个整数 $m, n$ ，以空格分隔，意义如题目所述；

接下来 $m$ 行，每行两个整数 $L_i, R_i$ ，以空格分隔，意义如题目所述（保证 $1 \leq L_i \leq R_i \leq n$ ，*ljh*会量子波动速读，所以他能在第 $i$ 个小时内看完任意页数）。

### 输出

第一行，如果*ljh*能看完这 $n$ 页复习资料，那么输出 **Yes**，否则输出 **No**。

第二行，输出*ljh*看的最多的页数 $x$ 。

### 输入样例1

```
5 10
1 5
5 6
5 8
5 9
10 10
```

## 输出样例1

```
Yes
5
```

## 输入样例2

```
4 8
1 2
2 3
3 4
4 5
```

## 输出样例2

```
No
2
```

## 样例解释

对于样例1，*ljh*成功看完了10页复习资料，所以输出 **Yes**，其中第5页看了4次，第6页看了3次，第7,8页看了2次，其余页数都只看了1次，所以 $x = 5$ ；

对于样例2，*ljh*只看了1到5页，并没能看完8页复习资料，所以输出 **No**，其中第2,3,4页都看了2次，其余都只看了1次，所以 $x$ 等于2,3,4中最小的那个，即 $x = 2$ 。

## 数据范围

$$1 \leq m, n \leq 1000, 1 \leq L_i \leq R_i \leq n$$

## HINT

观察数据范围我们发现 $m, n, L, R$ 的范围都很小，那么我们可以直接用一个数组来统计每页阅读的次数，假定这个数组为 $cnt$ ， $cnt[x]$ 表示第 $x$ 页阅读的次数。对于第 $i$ 个小时的复习计划，我们只需要让 $cnt[L_i]$ 到 $cnt[R_i]$ 全部+1就可以完成统计了，参考代码如下：

```
for(int i=1;i<=m;i++)
{
    int L,R;
    scanf("%d%d",&L,&R);
    for(int j=L;j<=R;j++)
        cnt[j]++;
}
```

统计完后问题就转化为查询一个数组中是否存在值为0的元素，以及查询数组中的最大值，相信这两个问题大家能很容易的解决。

AUTHOR: ljh

# C Meguru的最佳决策

“Ciallo~(∠ · ω)〰★”

## 题目描述

*Meguru* 在和超自研的大家一起玩游戏。其中有一关的任务是清除一块矩形区域内的敌人。目标区域有  $m$  行  $n$  列，即整个区域一共被分为  $m \times n$  块小区域，每块小区域内都有一定数量的敌人。*Meguru* 可以选择一块小区域发动技能，清除同一行和同一列上的全部敌人。现在她想知道，自己选择哪个位置的时候可以一次清除最多的敌人，最多可以清除多少。**保证只有一个位置**可以达到最多的清除数量。

## 输入

共  $m + 1$  行

第 1 行，两个正整数  $m, n$ ，以空格分隔，表示目标矩形区域的行数与列数

第 2 ~  $m + 1$  行，每行  $n$  个正整数，表示这一行中  $n$  块小区域内的敌人数量

## 输出

一行三个整数，以空格分隔，分别表示所选择发动技能的位置的行数、列数（都从 1 开始），以及所能清除的敌人的最大数量

## 输入样例

```
2 3
1 2 3
4 5 6
```

## 输出样例

```
2 3 18
```

## 数据范围

$1 \leq m, n \leq 400$ ,  $0 \leq$  每块小区域内的敌人数量  $\leq 10$

Author: Asahi

# D 回转吧！字符串！

## 题目描述

输入一个字符串  $s$ ，判断这个字符串首尾相连后，存在多少个长度为  $k$  的子串是回文串，回文串的定义是倒序后和原串完全一致的字符串。

注：比如字符串 "abcdb" 首尾相连的话，在长度  $k = 3$  时共有以下五个子串："abc" "bcd" "cdb" "dba" "bab"，显然只有最后一个是回文串，所以存在一个长度  $k = 3$  时的回文串。

## 输入

两行，第一行为输入的字符串  $s$ ，第二行为一个整数  $k$  含义如题。

## 输出

一行，一个整数代表有几个长度为  $k$  的子串是回文串。

## 样例数据

### 输入样例-1

```
aaaaa
2
```

### 输出样例-1

```
5
```

### 输入样例-2

```
acbdffdbca
4
```

### 输出样例-2

```
2
```

## 数据范围

对于 100% 的数据，满足  $1 \leq k \leq |s| \leq 10^3$ ，保证字符串中只包含小写字母。其中  $|s|$  代表字符串  $s$  所包含的字符个数。

## HINT

字符串最后还需要一个位置存 `"\0"`，所以记得数组开大点。

Author: Untitled

# E ✿ 樱花满地集于我心 ✿

## 题目描述

在《罪恶王冠》所有故事尘埃落定之后，樱满集每天都会拄着拐杖来到湖边。

湖面上有很多的小岛，看到这些小岛，集想到了会让人晶体化的默示录病毒，也想起了和楪祈还有伙伴一起战斗的时光。

默示录病毒的作用机制非常奇特，它们总是三个同时产生作用，相互感应。

默示录病毒有两种作用面积，是平均作用面积和最小作用面积。

**平均作用面积指这三个默示录病毒围成的三角形的面积。**

**最小作用面积指上述三角形的内切圆的面积。**

湖里面有  $n$  个小岛，樱满集会联想  $m$  次，每次只会想到这三个岛的一种作用面积。

他想请你帮帮他，算出这些作用面积

## 输入

输入共  $n + m + 2$  行

第一行：一个整数  $n$ ，代表小岛（二维平面上的点）的数量

后续  $n$  行：每行两个整数  $x, y$ ，代表一个小岛的坐标，小岛的编号按输入顺序从 1 到  $n$

第  $n + 2$  行：一个整数  $m$ ，代表樱满集联想的次数

后续  $m$  行，每行有四个整数，前三个整数分别代表樱满集联想到的三个小岛的编号

最后一个整数  $opt$

- $opt==1$ ：求出平均作用面积
- $opt==2$ ：求出最小作用面积

按照 $pi = 3.1415926$ 计算

如果这三个小岛无法构成三角形，则输出一行 Yuzuriha Inori

(因为樱满集对楪祈思念过度，才会出现这样的情况)

## 输出

输出共 $m$ 行，每一次在询问之后保留三位小数输出计算结果或者 Yuzuriha Inori

## 样例

### 样例输入

```
4
(1,9)
(2,6)
(0,8)
(1,7)
1
1 2 4 2
```

### 样例输出

```
0.291
```

## 数据范围

$$3 \leq n \leq 1000$$

$$1 \leq m \leq 5000$$

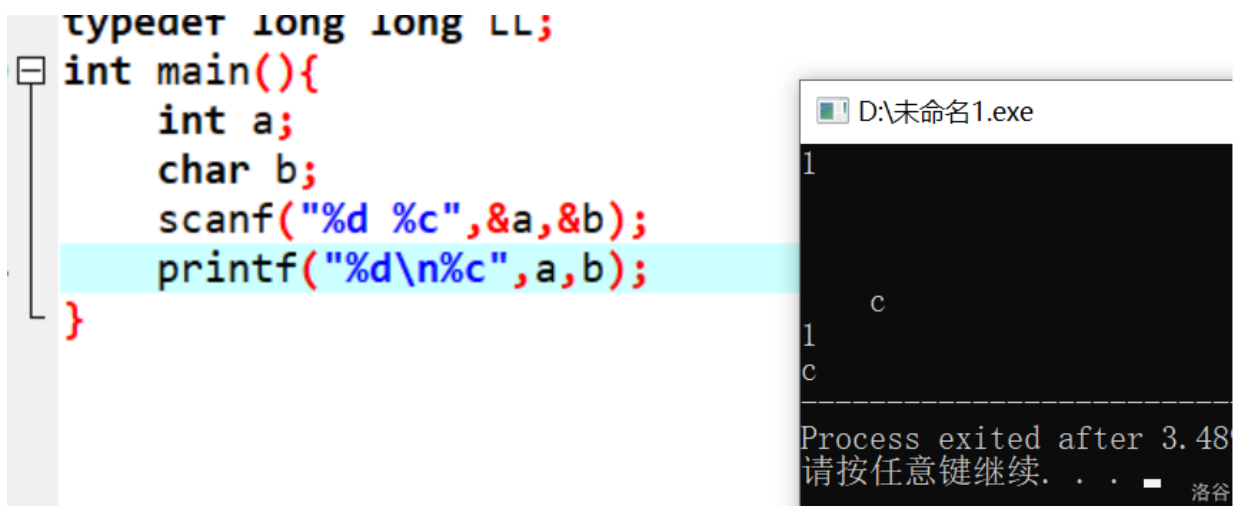
$$-1000 \leq x, y \leq 1000$$

保证不会出现坐标相同的小岛

保证不会选取编号相同的小岛计算

## Hint

1. 有一个可以用于计算三角形面积的公式叫海伦公式，设三角形三边长度为 $a$ 、 $b$ 、 $c$ ，令 $p = (a + b + c)/2$ ，则三角形的面积 $S = \sqrt{p \cdot (p - a) \cdot (p - b) \cdot (p - c)}$ ，或者可以通过行列式计算叉乘
2. 三角形的内切圆半径计算方法： $r = 2S/(a + b + c)$ ，其中 $a, b, c$ 为三角形三边， $S$ 为三角形面积
3. 读入时可以使用一个空格来匹配所有的空白符，请观察如下代码：



P.S.

✧樱花满地集于我心✧蝶舞纷飞祈愿相随✧

author: void

## G 朝田诗乃的朋友圈

### 题目描述

朝田诗乃不仅是 GGO 中有名的冰之狙击手，也是一个热爱分享生活的女孩子。

这天，她和往常一样刷着朋友圈，突然好奇列表里每个人能看到几条朋友圈。

为了便于解决这个问题，诗乃进行如下规定：

现在她的好友列表里一共有  $n$  个好友（编号依次为从 1 到  $n$ ），在一段时间里有  $m$  个按时间顺序排列的消息记录：

- !  $x$  表示用户  $x$  发了一条朋友圈
- +  $x$   $y$  表示用户  $x$  和  $y$  成为了好友
- $x$   $y$  表示用户  $x$  和  $y$  解除了好友关系

当一个用户发朋友圈时，假设  $ta$  所有的好友都能看见  $ta$  的消息

现在，诗乃给出了这个操作序列，假设最开始所有人都不是好友关系，且操作序列完全合法（即 +  $x$   $y$  时  $x$  和  $y$  一定不是好友，而 -  $x$   $y$  时  $x$  和  $y$  一定是好友）。

现在，请你编写程序回答诗乃提出的问题吧。



## 输入

第 1 行 2 个整数  $n, m$ 。

接下来  $m$  行，按时间顺序读入  $m$  条记录，每条记录的格式如题目所述，用空格隔开。

## 输出

输出一行  $n$  个用空格隔开的数，第  $i$  个数表示用户  $i$  最后看到了几条消息。

## 输入样例

```
2 8
! 1
! 2
+ 1 2
! 1
! 2
- 1 2
! 1
! 2
```

## 输出样例

```
1 1
```

##样例解释

只有第 4 和第 5 条记录对应的消息被看到过。其他消息发送时，1 和 2 不是好友。

##数据范围

$n \leq 1000, m \leq 10000$

# H 哪吒的区间合并（1）

## 题目描述

给出若干个端点为正整数的闭区间，请进行\*\*区间合并\*\*，将这些区间合并为\*\*不相交\*\*的闭区间。

## 输入

不定行输入，每行两个空格隔开的正整数  $left, right$ ，表示区间  $[left, right]$

## 输出

输入若干行，表示合并后的区间

每行包含两个空格隔开的正整数  $left, right$ ，表示区间  $[left, right]$ ，各个区间请按升序排列输出。

## 样例

### 输入

```
1000 6666
114514 810975
6666 12345
555 999
5 666
100000 100002
```

### 输出

```
5 999
1000 12345
100000 100002
114514 810975
```

## 数据范围

输入区间个数不超过  $10^6$  个

$$1 \leq left < right \leq 10^6$$

## Hint

可以用数组存储区间，每次读入区间，左端点为数组元素的下标，右端点为数组元素的值，最后遍历数组进行合并。

每次读入区间，也可以标记数组下标分别为区间左端点和右端点的元素，最后遍历一遍输出区间。

Author: 哪吒

# I 繁琐的代码

## 题目背景

怠惰且笨拙的Uanu写的代码非常繁琐。当协作时，别人都抱怨看不懂Uanu的代码，着急的Uanu找到了你，想让你帮他把代码简化下！

## 题目描述

现在你需要处理一份**繁琐**的C语言代码，你需要对其进行如下简化：

- 删除代码中的注释：
  - 单行注释：以 `//` 开始，以换行符结束，**不包括换行符**。
  - 多行注释：以 `/*` 开始，以 `*/` 结束。

如有疑问，请一定要看数据说明。

## 输入格式

不定行输入，即一段C语言编写的程序代码，长度不超过1000行。

## 输出格式

不定行输出，即简化后的代码。

## 输入样例

```
#include <stdio.h>
int a = 0; //Define variable

int main() { /*Define function*/a=1;
    goto label;
    printf("%d", a); //Output
    label:
    return 0;
}
```

## 输出样例

```
#include <stdio.h>
int a = 0;

int main() { a=1;
    goto label;
    printf("%d", a);
    label:
    return 0;
}
```

## 数据说明

- 字符常量和字符串字面量中不包含转义字符 `\'` 和 `\"`
- 每行字符数不超过200
- 保证引号成对出现
- C语言中，字符常量和字符串字面量中出现的 `//` 或 `/*` 不会被视为注释的开始
- 输入中未被注释的空白符（空格符，制表符，换行符）应当被保留，结尾无换行
- 无需考虑注释内的内容和简化后程序的正确性

author: Uanu

# J 鼠标坏啦

## 题目描述

Jerydeak 的鼠标坏了，但他要输入  $n$  个字符 `0`，他可以进行以下操作：

- 输入 `0`；
- `Ctrl+A`，进入全选状态；
- `Ctrl+C`，复制全部内容（在全选状态才可以使用）；
- `Ctrl+V`，输入复制的内容；

注意，每次输入时，如果在全选状态，会将用输入**覆盖**当前内容并退出全选状态。

Jerydeak 的初始文档是空白的，请问他最少需要几步操作，才能输入**恰好**  $n$  个 `0`？

## 输入格式

不定行输入，每一行一个整数，表示一组输入。

## 输出格式

对于每组输入，输出一行整数，表示最少的操作数。

## 样例

### 输入 #1

```
0
2
12
```

### 输出 #1

```
0
2
9
```

## 数据范围

数据组数不超过  $10^3$ ，且对每组数据  $0 \leq n \leq 10^3$ 。