基础算法B

前缀和与差分

Mas

一维前缀和

前缀和是一种重要的预处理方式

可以简单理解为“数组的前 n 项的和”

$$sum_i = \sum_{j=1}^{i} a_j$$

$$= sum_{i-1} + a_i$$

若有前缀和数组sum

$$a_l + a_{l+1} + \ldots + a_{r-1} + a_r$$

$$= sum_r - sum_{l-1}$$

可将单次区间求和的复杂度降至 $O(1)$

	1	2	3	4	5	6	7	8	9	10
arr	3	2	4	1	6	5	10	8	7	9
sum	3	5	9	10	16	21	31	39	46	55

#1459 求和

题目描述

给你一个长为 N 的 int 数组,进行 Q 次询问

每次询问给出 L 和 R

请对每次询问输出 $a_L + a_{L+1} + \cdots + a_R$ 的和

输入格式

第一行 N ,第二行 N 个数,第三行 Q ,接下来每行一个询问 L 和 R

输出格式

对于每次询问输出对应的答案

数据范围

对于全部数据 $1 \leq N, Q \leq 100000, a_i \leq 10000$

输入样例1

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

输出样例1

```
3
12
9
15
```

#1849 m子段和

题目描述

给出一个长度为 n 的序列 a ，选出其中连续且长度为 m 的一段使得这段和最大。

输入格式

第一行是两个个整数 $n(1 \leq n \leq 10^5), m(1 \leq m < n)$ 。

第二行有 n 个整数，第 i 个整数表示序列的第 i 个数字 $a_i(-10^9 \leq a_i \leq 10^9)$

输出格式

输出一行一个整数表示答案。

输入样例

```
7 3
2 -4 3 -1 2 -4 3
```

输出样例

```
4
```

#1848 前缀和的逆

题目描述

有 n 个正整数放到数组 B 里，它是数组 A 的前缀和数组，求 A 数组。

输入格式

第一行 1 个正整数 n 。

第二行 n 个正整数 。

输出格式

n 个正整数。

输入样例

```
6
2 10 20 25 30 43
```

输出样例

```
2 8 10 5 5 13
```

说明/提示

对于 100% 的数据，满足 $N \leq 100$、$B_i \leq 10000$ 。

#2359、和为K的子数组

题目描述

给定一个长度为 n 的整数数组 A 和一个整数 k

你需要找到该数组中和为 k 的连续的子数组(长度至少为 1)的个数

输入格式

第一行两个正整数 n, k

第二行 n 个整数 a_i 表示数组的每个元素

输出格式

输出和为 k 的连续的子数组的个数

输入样例

```
3 2
1 1 1
```

输出样例

```
2
```

样例解释

下标 $[1,2]$ 与 $[2,3]$ 为两种不同的情况

数据范围

对于 20% 的数据 $1 \le n \le 100$

对于 100% 的数据 $1 \le n \le 200000$

对于全部的数据 $-10^9 \le k \le 10^9, -20000 \le a_i \le 20000$

计算出前缀和数组sum_i

需要统计$sum_i - sum_j = k$的下标二元组数量

仅需要统计$sum_i - k$的数量即可

时间复杂度$O(n)$

一维前差分

差分是前缀和相对的策略,可看作前缀和的逆运算

令 $d_i = a_i - a_{i-1}$,即相邻两数的差

$$d_1 + d_2 + \cdots + d_i$$

$$= a_1 + a_2 - a_1 + \cdots + a_i - a_{i-1}$$

对d做一遍前缀和就得到了原数组a,即

$$a_i = \sum_{j=1}^{i} d_j$$

对于前缀和sum_i

$$sum_i = \sum_{j=1}^{i} a_j = \sum_{j=1}^{i} \sum_{k=1}^{j} d_k = \sum_{j=1}^{i} (i - j + 1) \times d_j$$

#1460 加值

题目描述

给你一个长为 N 的 int 数组 a ,进行 Q 次操作

每次操作给出 L 和 R ,表示在 $a_l, a_{l+1}, \ldots, a_{r-1}, a_r$ 上每个数加 1

请输出 Q 次操作后数组 a

输入格式

第一行 N ,第二行 N 个数,第三行 Q ,接下来每行一个操作 L 和 R

输出格式

输出最后的数组 a

数据范围

对于全部的数据 $1 \leq N \leq Q \leq 100000$

输入样例1

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

输出样例1

```
3 8 5 7 5
```

#1460 加值

多次对序列的一个区间加上一个数,如令$a[l] \sim a[r]$ 每个数加上一个 k

只需要让 $d[l] \longrightarrow d[l]+k, d[r+1] \longrightarrow d[r+1]-k$

最后做一遍前缀和求出a数组即可

对于一次加值操作时间复杂度为$O(1)$

	1	2	3	4	5
arr	1	2	3	4	5
d	1	1	1	1	1

	1	2	3	4	5
arr	1	3	4	5	5
d	1	2	1	1	0

#378 工厂流水线

题目描述

SYC 工厂需要生产 n 个产品

每个产品会在记录本上记录开始生产的时间 x 以及完成生产的时间 y

现在 Mas 拿到这本记录本以后想知道最多有多少件产品同时在生产线上生产

> *在同一时刻总是开始生产的产品先进入流水线*

输入格式

输入第一行只有一个整数 n ,表示记录本上共记录了 n 件产品的信息

接下来 n 行,每行两个整数 x 和 y ,表示一件产品开始生产的时间和完成生产的时间

输出格式

输出仅有一行,该行只有一个整数,表示最多有多少件产品同时在生产线上生产

数据范围

对于 50% 的数据中, $1 \leq n \leq 1000, 1 \leq x, y \leq 1000$

对于 100% 的数据中, $1 \leq n \leq 100000, 1 \leq x, y \leq 100000000$

样例输入

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

样例输出

```
3
```

#378 工厂流水线

思路1

区间加定值值,借助差分数组优化

时间复杂度$O(n)$

空间复杂度$O(\max(y))$

思路2

将每个产品拆分成左右端点,对端点的位置进行排序

从前往后扫描所有端点

遇到左端点计数器 $+1$

遇到右端点计数器 -1

时间复杂度 $O(n\log n)$

空间复杂度 $O(n)$

#1850、最高的牛

题目描述

有 N 头牛站成一行，被编队为 1、2、$3...N$，每头牛的身高都为整数。

当且仅当两头牛中间的牛身高都比它们矮时，两头牛方可看到对方。

现在，我们只知道其中最高的牛是第 P 头，它的身高是 H，剩余牛的身高未知。

但是，我们还知道这群牛之中存在着 M 对关系，每对关系都指明了某两头牛 A 和 B 可以相互看见。

求每头牛的身高的最大可能值是多少。

输入格式

第一行输入整数 N，P，H，M，数据用空格隔开。

接下来 M 行，每行输出两个整数 A 和 B，代表牛 A 和牛 B 可以相互看见，数据用空格隔开。

输出格式

一共输出 N 行数据，每行输出一个整数。

第 i 行输出的整数代表第 i 头牛可能的最大身高。

数据范围

对于全部数据 $1 \le N \le 10000, 1 \le H \le 1000000, 1 \le A, B \le 10000, 0 \le M \le 10000$

> *注意：此题中给出的关系对可能存在重复*

输入样例

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

输出样例

```
5
4
5
3
4
4
5
5
5
```

#1850、最高的牛

若a和b之间可以相互看到不妨设身高都为h

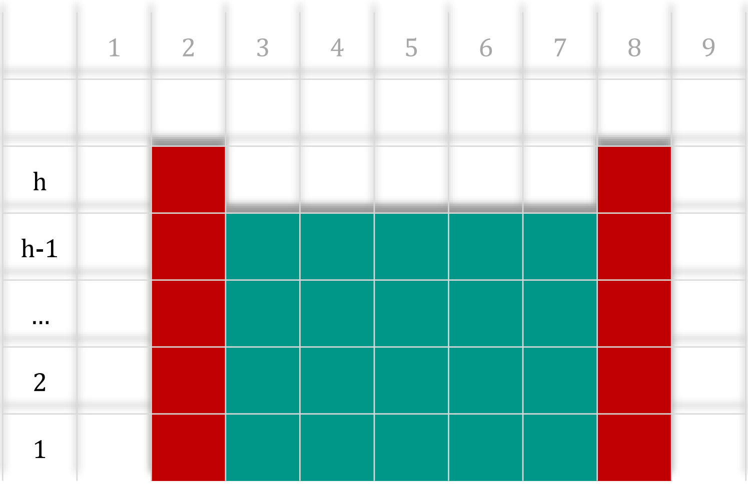
那么$[a+1,b-1]$范围内所有牛的身高至多为$h-1$

假设最开始所有牛的身高都为H

对所有可见关系去重

使用差分数组维护$[a+1,b-1]$范围的牛的身高

时/空间复杂度$O(n)$



二维前缀和

定义sum为a前缀和数组,$sum_{i,j}$表示 a 数组前i行前j列所有元素之和

$$sum_{x,y} = \sum_{i=1}^{x}\sum_{j=1}^{y} a_{i,j}$$

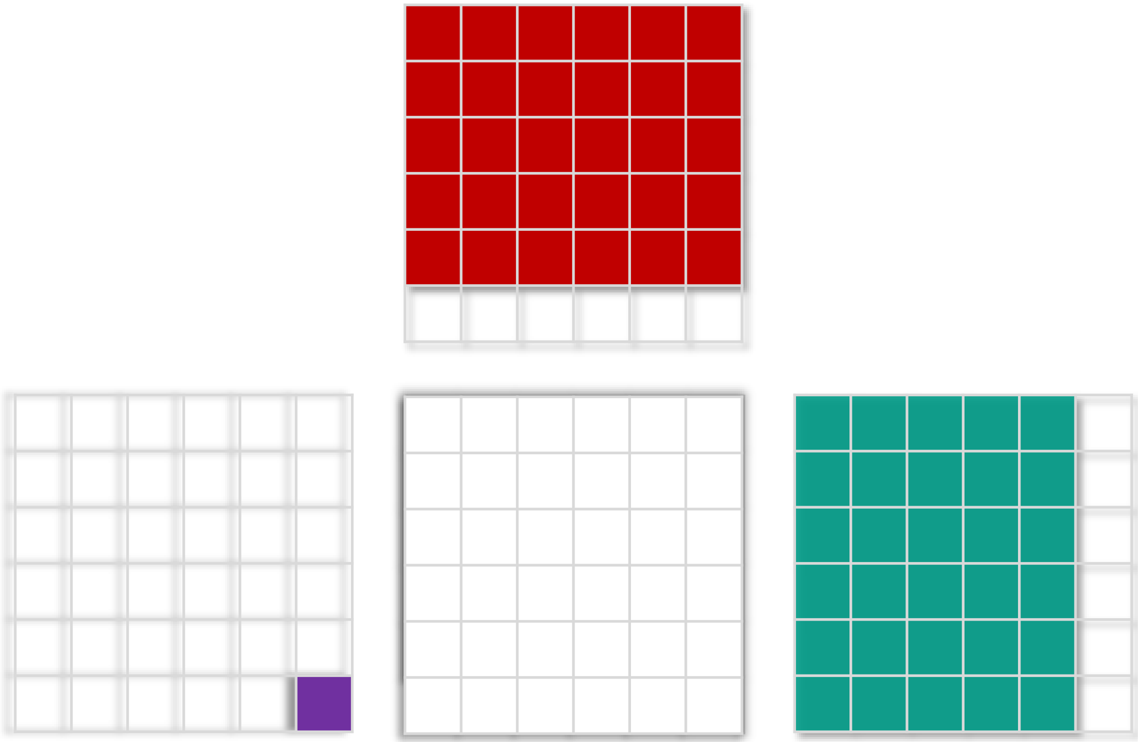
递推式如下

$$sum_{i,j} = sum_{i-1,j} + sum_{i,j-1} + a_{i,j} - sum_{i-1,j-1}$$

若给定子矩阵左上角坐标(x_1, y_1)右下角坐标(x_2, y_2)

子矩阵的和为

$$sum_{x_2,y_2} - sum_{x_2,y_1-1} - sum_{x_1-1,y_2} + sum_{x_1-1,y_1-1}$$



#1851 二维前缀和

题目描述

求一个 $n \times m$ 大小的二维矩阵对应的前缀和。

输入

第一行 2 个正整数：N 和 M，N 和 M 范围在 $[1, 1000]$。
其后 n 行，每行 M 个正整数：范围在 $[1, 10000]$。

输出

对应二维数组的前缀和。

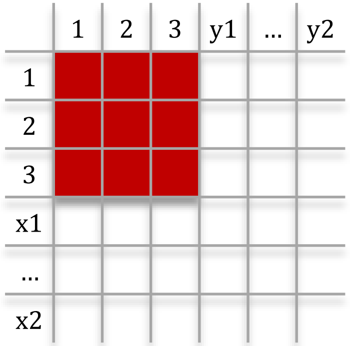
样例输入

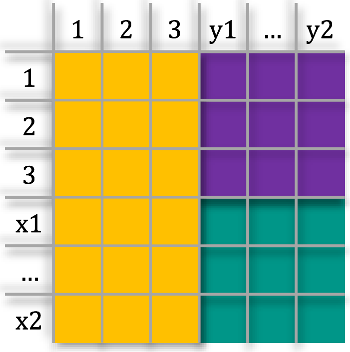
```
3 4
1 2 4 3
5 1 2 4
6 3 5 9
```

样例输出

```
1 3 7 10
6 9 15 22
12 18 29 45
```

子矩阵和





要求子矩阵$a[\,x1 \sim x2\,][\,y1{\sim}y2\,]$的和

黄色部分： $sum[x2][y1-1]$

紫色部分： $sum[x1-1][y2]$

红色部分： $sum[x1-1][y1-1]$

所求子矩阵和为

$$sum[x2][y2]\ -\ sum[x2][y1-1]\ -\ sum[x1-1][y2]\ +\ sum[x1-1][y1-1]$$

#1852 子矩阵的和

题目描述

输入一个 n 行 m 列的整数矩阵

再输入 q 个询问

每个询问包含四个整数 x_1, y_1, x_2, y_2 ,表示一个子矩阵的左上角坐标和右下角坐标

对于每个询问输出子矩阵中所有数的和

输入

第一行包含三个整数 n, m, q

接下来 n 行,每行包含 m 个整数,表示整数矩阵

接下来 q 行,每行包含四个整数 x_1, y_1, x_2, y_2 ,表示一组询问

输出

共 q 行,每行输出一个询问的结果

数据范围

对于全部的数据 $1 \le n, m \le 1000, 1 \le q \le 200000, 1 \le x_1 \le x_2 \le n, 1 \le y_1 \le y_2 \le m, -1000 \le a_{i,j} \le 1000$

样例输入

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

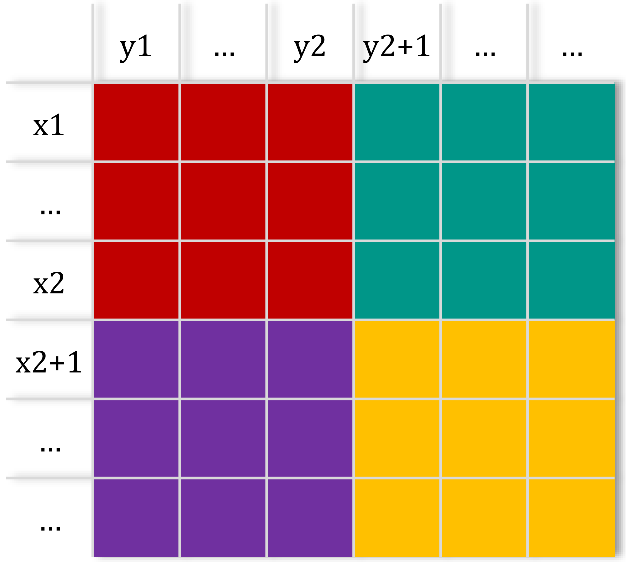
样例输出

```
17
27
21
```

二维差分

要使$a[\,x1 \sim x2\,][\,y1{\sim}y2\,]$区域的元素加上定值$k$,只需

- $d[x1][y1] \mathrel{+}= k$
- $d[x1][y2+1] \mathrel{-}= k$
- $d[x2+1][y1] \mathrel{-}= k$
- $d[x2+1][y2+1] \mathrel{+}= k$



- $d[x1][y2+1] \mathrel{-}= k$

抵消$d[x1][y1] \mathrel{+}= k$后续影响

- $d[x2+1][y2+1] \mathrel{+}= k$

抵消$d[x2+1][y1] \mathrel{-}= k$后续影响

- $d[x2+1][y1] \mathrel{-}= k$

抵消$d[x1][y1] \mathrel{+}= k$后续影响

#2041 差分矩阵

题目描述

输入一个 n 行 m 列的整数矩阵 A ,再输入 q 个操作,每个操作包含五个整数 x_1, y_1, x_2, y_2, c ,其中 (x_1, y_1) 和 (x_2, y_2) 表示一个子矩阵的左上角坐标和右下角坐标

每个操作都要将选中的子矩阵中的每个元素的值加上 c

请你将进行完所有操作后的矩阵输出

输入

第一行包含整数 n , m , q
接下来 n 行,每行包含 m 个整数,表示整数矩阵
接下来 q 行,每行包含 5 个整数 x_1, y_1, x_2, y_2, c ,表示一个操作

输出

共 n 行,每行 m 个整数,表示所有操作进行完毕后的最终矩阵

数据范围

对于 100% 的数据 $1 \le n, m \le 1000, 1 \le q \le 100000, 1 \le x_1 \le x_2 \le n, 1 \le y_1 \le y_2 \le m, -1000 \le c \le 1000, -1000 \le a_{ij} \le 1000$

样例输入

```
3 4 3
1 2 2 1
3 2 2 1
1 1 1 1
1 1 2 2 1
1 3 2 3 2
3 1 3 4 1
```

样例输出

```
2 3 4 1
4 3 4 1
2 2 2 2
```

LAB

实验舱

青少年编程

走近科学 走进名校

谢谢观看