

NOI2013 浙江省组队选拔赛试题

第一试

题目名称	K 大数查询	蚂蚁寻路	防守战线
可执行文件名	sequence	ant	defend
输入文件名	sequence.in	ant.in	defend.in
输出文件名	sequence.out	ant.out	defend.out
每个测试点时限	2	2	2
内存限制	512M	512M	512M
测试点数目	10	10	10
每个测试点分值	10	10	10
是否有部分分	否	否	否
题目类型	传统型	传统型	传统型
是否有附加文件	无	无	无

提交源程序须加后缀

对于 C++ 语言	sequence.cpp	ant.cpp	defend.cpp
对于 C 语言	sequence.c	ant.c	defend.c
对于 Pascal 语言	sequence.pas	ant.pas	defend.pas

注意：最终测试时，所有编译命令均不打开任何优化开关。

K 大数查询

【问题描述】

有 n 个位置和 m 个操作。操作有两种，每次操作如果是 $1\ a\ b\ c$ 的形式，表示往第 a 个位置到第 b 个位置每个位置加入一个数 c 。如果操作形如 $2\ a\ b\ c$ 的形式，表示询问从第 a 个位置到第 b 个位置，第 c 大的数是多少。

【输入格式】

在输入文件 *sequence.in* 中，第一行两个数 n, m 。意义如题目描述。
接下来 m 行每行形如 $1\ a\ b\ c$ 或者 $2\ a\ b\ c$ 如题目描述。

【输出格式】

在输出文件 *sequence.out* 中，对于每个询问回答 k 大数是多少。

【样例输入】

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

【样例输出】

```
1
2
1
```

【样例说明】

第一个操作后位置 1 的数只有 1，位置 2 的数也只有 1。第二个操作后位置 1 的数有 1、2，位置 2 的数也有 1、2。第三次询问位置 1 到位置 1 第 2 大的数是 1。第四次询问位置 1 到位置 1 第 1 大的数是 2。第五次询问位置 1 到位置 2 第 3 大的数是 1。

【数据规模与约定】

30%的数据 $n=m=1000$

100%的数据 $n, m \leq 50000$ ，并且后 7 个点的数据 n, m 的范围从 32000 到 50000
近似成等差数列递增。 $a \leq b \leq n, 1$ 操作中 $|c| \leq n, 2$ 操作中 $|c| \leq \text{maxlongint}$

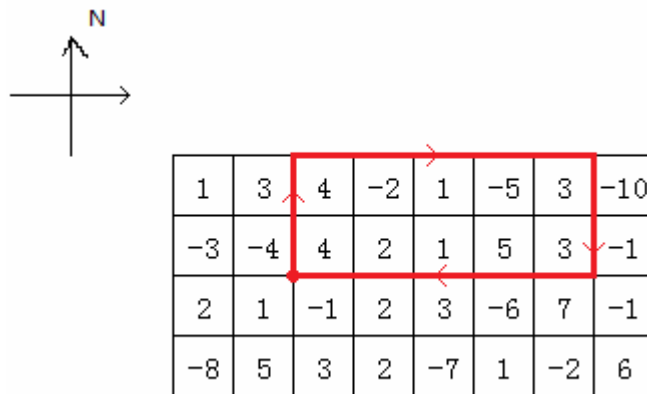
蚂蚁寻路

【问题描述】

在一个 $n*m$ 的棋盘上，每个格子有一个权值，初始时，在某个格子的顶点处一只面朝北的蚂蚁，我们只知道它的行走路线是如何转弯，却不知道每次转弯前走了多长。蚂蚁转弯是有一定特点的，即它的转弯序列一定是如下的形式：

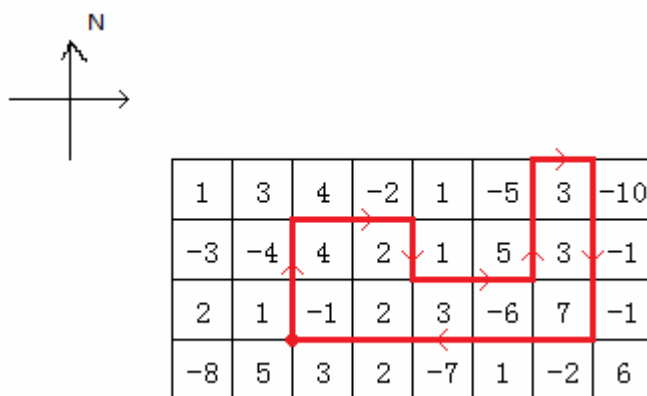
右转，右转，左转，左转，右转，右转…左转，左转，右转，右转，右转。
即两次右转和两次左转交替出现的形式，最后两次右转（最后两次一定是右转）后再多加一次右转。我们还知道，蚂蚁不会在同一个位置连续旋转两次，并且蚂蚁行走的路径除了起点以外，不会到达同一个点多次，它最后一定是回到起点然后结束自己的行程，而且蚂蚁只会在棋盘格子的顶点处转弯。

设 k 为蚂蚁左转的次数除以2，当 $k=0$ 时，蚂蚁可能行走的路径如下图：



转弯序列为：右转，右转，右转。

当 $k=1$ 时，蚂蚁可能行走的路径如下图：



转弯序列为：右转，右转，左转，左转，右转，右转，右转。

现在已知棋盘大小、每个格子的权值以及左转次数/2 的值，问蚂蚁走出的路径围出的封闭图形，权值之和最大可能是多少。

【输入格式】

在输入文件 *ant.in* 中，第一行三个数 n,m,k 。意义如题目描述。
接下来一个 n 行 m 列的整数矩阵，表示棋盘。

【输出格式】

在输出文件 *ant.out* 中，一个数，表示蚂蚁所走路径围出的图形可能的最大权值和。

【样例输入】

```
2 5 2
-1 -1 -1 -1 -1
-1 -1 -1 -1 -1
```

【样例输出】

```
-8
```

【样例说明】

除了第一行的第二个和第一行的第四个都要围起来才至少合法。

【数据规模与约定】

10%的数据所有格子中权值均非负

另 20%的数据 $n=2$

另 30%的数据 $k=0$

100%的数据 $1 \leq n \leq 100, 1 \leq m \leq 100, 0 \leq k \leq 10$ 保证存在合法路径，数据有梯度，格子中每个元素的值绝对值不超过 10000

防守战线

【题目描述】

战线可以看作一个长度为 n 的序列，现在需要在这个序列上建塔来防守敌兵，在序列第 i 号位置上建一座塔有 C_i 的花费，且一个位置可以建任意多的塔，费用累加计算。有 m 个区间 $[L_1, R_1]$, $[L_2, R_2]$, \dots , $[L_m, R_m]$ ，在第 i 个区间的范围内要建至少 D_i 座塔。求最少花费。

【输入格式】

输入文件 *defend.in* 中，第一行为两个数 n, m 。
接下来一行，有 n 个数，描述 C 数组。
接下来 m 行，每行三个数 L_i, R_i, D_i ，描述一个区间。

【输出格式】

输出文件 *defend.out* 中，仅包含一行，一个数，为最少花费。

【输入样例】

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

【输出样例】

```
11
```

【样例说明】

位置 1 建 2 个塔，位置 3 建一个塔，位置 4 建一个塔。花费 $1*2+6+3=11$ 。

【数据规模】

对于 20% 的数据， $n \leq 20$, $m \leq 20$ 。
对于 50% 的数据（包括上部分的数据）， D_i 全部为 1。
对于 70% 的数据（包括上部分的数据）， $n \leq 100$, $m \leq 1000$ 。
对于 100% 的数据， $n \leq 1000$, $m \leq 10000$, $1 \leq L_i \leq R_i \leq n$ ，其余数据均 ≤ 10000 。