

信息学竞赛联赛模拟赛

November 2020

Linyi No.1 Middle School

(请选手务必仔细阅读本页内容)

一、题目概况

中文题目名称	数论题	模拟题	动态规划题	贪心题
子目录名	theory	simulate	dynamic	greedy
输入文件名	theory.in	simulate.in	dynamic.in	greedy.in
输出文件名	theory.out	simulate.out	dynamic.out	greedy.out
每个测试点时限	1 秒	1 秒	1 秒	1 秒
内存上限	512M	512M	512M	512M
测试点数目	10	20	25	?
每个测试点分值	10	5	4	?
附加样例文件	有	有	有	有
结果比较方式	全文比较（过滤行末空格及文末回车）			
题目类型	传统	传统	传统	传统

二、提交源程序程序名

对于 C++ 语言	theory.cpp	simulate.cpp	dynamic.cpp	greedy.cpp
-----------	------------	--------------	-------------	------------

三、优化开关

对于 C++ 语言	-O2 -lm -std=c++11	-O2 -lm -std=c++11	-O2 -lm -std=c++11	-O2 -lm -std=c++11
-----------	-----------------------	-----------------------	-----------------------	-----------------------

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()` 的返回类型必须是 `int`，程序正常结束时的返回值必须是 0。
3. 最终评测机器 CPU 为 i5-9300H@ 2.40GHz，内存 16G，使用 Lemonlime 评测。
4. 此次比赛需要建立子文件夹，不保证难度递增排序，不保证题目名称与题目有关，请选手合理安排做题时间。

Problem A. 数论题

Input file: theory.in
Output file: theory.out
Time limit: 1 second
Memory limit: 512 megabytes

小 L 沉迷一款名叫明日非洲的塔防游戏无法自拔。

明日非洲的地图可以抽象成一条长度为 n 的数轴，其有 $n + 1$ 个端点。对于 $i \in [1, n]$ 的所有整数，区域 $[i - 1, i]$ 上存在一个战斗力为 s_i 的战士。小 L 扮演的角色 Amiya 可以任意移动到一个 $n + 1$ 个端点中的其中一个，设其为 x ，那么他可以指挥 x 左侧 d 个战士以及右侧 d 个战士（如果存在的话）。由于 Amiya 杰出的领袖气质，她可以使得被指挥的这些战士的战斗力变为 $s_i(d - k)^2$ ，其中 k 为第 i 个战士和 Amiya 之间相隔的战士个数。

现在小 L 想知道被 Amiya 指挥的战士的战斗力之和最大是多少。

Input

第一行两个整数 n, d 表示地图的大小和技能的半径。

第二行 n 个整数，第 i 个表示 s_i 。

Output

一行，一个整数，表示最大的战斗力之和。

Examples

theory.in	theory.out
5 2 1 1 3 5 6	47
见选手目录下 theory/theory2.in	见选手目录下 theory/theory2.ans

Scoring

对于 10% 的数据， $n \leq 5$ 。

对于 40% 的数据， $n \leq 2 \times 10^3$ 。

对于 60% 的数据， $n \leq 10^5$ 。

另有 10% 的数据， $s_i = 1$ 。

对于 100% 的数据， $1 \leq n \leq 10^6, 1 \leq d \leq 3 \times 10^4, 1 \leq s_i \leq 10^3$

Problem B. 模拟题

Input file: `simulate.in`
Output file: `simulate.out`
Time limit: 1 second
Memory limit: 512 megabytes

众所周知，签到题应该是一道模拟题。

NOI 失利回家打工的小 W 和小 Y 被安排了加工小球的任务，现在有 10^{100} 个小球排在他们面前，他们要按照自己的方式给小球上色。小球的编号从 0 开始。

想完成这个工作需要两个人配合，小 W 拿到了有 a 个颜色的油漆桶，小 Y 拿到了有 b 个颜色的油漆桶，颜色的编号也从 0 开始。

第一个人需要给小球上色。假设第一个人有 x 种颜色，他会从第一个球开始，循环地给所有小球上色。（即，0 号球的颜色为 0，1 号球的颜色为 1， \dots ， x 号球的颜色为 0， $x+1$ 号球的颜色为 1， \dots ）。

第二个人需要将所有相同颜色的小球装在一个箱子里，显然现在有 x 个箱子（没有小球的箱子也算），箱子编号从 0 开始（标号为 x 的箱子里面装了所有颜色为 x 的球）。假设第二个人有 y 种颜色，他会从第一个箱子开始，循环的给所有箱子上色。（即，0 号箱子的颜色为 0，1 号箱子的颜色为 1， \dots ， y 号箱子的颜色为 0， $y+1$ 号箱子的颜色为 1， \dots ）

有时候会由小 W 给小球上色，有时候也会由小 Y 给小球上色，现在黑心老板小 H 想知道，编号在 $[l, r]$ 内有多少个球，使得无论是谁给小球上色，它所在的箱子的颜色都是相同的。

注意：我们只关心小球所在的箱子的颜色是否相同，而不关心箱子的编号或是小球的编号，请仔细阅读样例解释确保你理解了题意。

Input

第一行一个正整数 T 表示数据组数。

接下来 T 行，每行四个正整数 a, b, l, r ，表示一组询问。

Output

T 行，表示每次询问的答案。

Examples

<code>simulate.in</code>	<code>simulate.out</code>
4	2
3 4 1 5	191
20 30 114 514	318822
23 45 123123 456456	193421756
12345 67890 10292834 203918259	
见选手目录下 <code>simulate/simulate2.in</code>	见选手目录下 <code>simulate/simulate2.ans</code>

Notes

对于样例 1:

如果小 W 给小球染色, 小球的颜色分别是 1, 2, 0, 1, 2, 小球所在箱子的颜色是 1, 2, 0, 1, 2。

如果小 Y 给小球染色, 小球的颜色分别是 1, 2, 3, 4, 0, 所在箱子的颜色是 1, 2, 3, 0, 1。

不难发现只有 0 号小球, 1 号小球满足条件。

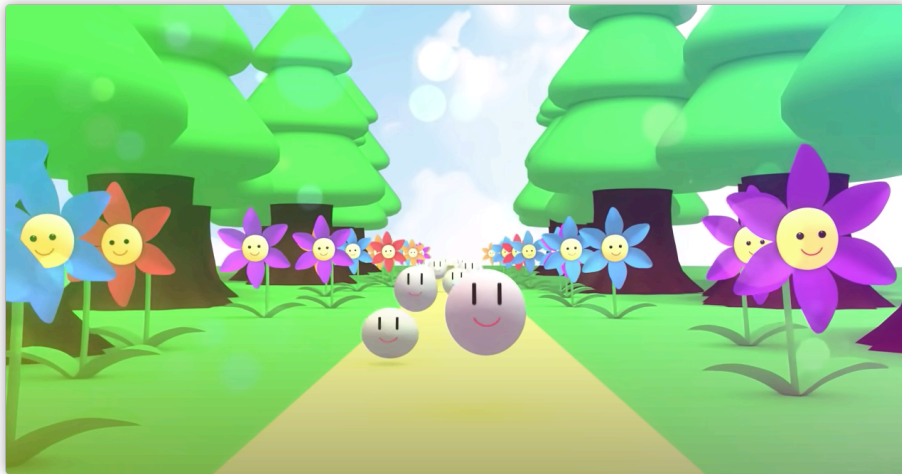
Scoring

对于 20% 的数据, 有 $1 \leq a, b \leq 50, 1 \leq l, r \leq 10^3, T \leq 10$ 。

对于 50% 的数据, 有 $1 \leq a, b \leq 50, T \leq 10^3$ 。

对于 100% 的数据, 有 $1 \leq a, b \leq 10^9, 1 \leq l \leq r \leq 10^{16}, 1 \leq T \leq 10^5$ 。

有人问我小球长什么样, 我可以告诉你是这样的:



Problem C. 动态规划题

Input file: dynamic.in
Output file: dynamic.out
Time limit: 1 second
Memory limit: 512 megabytes

小 G 最近沉迷《只狼：白给二度》无法自拔。

小 G 操作的角色狼来到了热情好客的水生村，水生村中住着 n 个村民。

水生村中有无数只能住一个村民的房子。狼为了报答热情好客的村民，为他们建造了 m 种房子，每种房子有 a_i 个，每个房子最多可以住 b_i 个村民。

知恩图报的狼忘不了村民的热情，于是他每年都会来水生村进行一些改造，持续 q 年。这 q 年中，狼只会进行修建和拆除其中一种改造，也就是说这 q 年狼要么都每年都来修建一些房子，要么每年都来拆除一种房子。

每个村民感受到的温暖度为他所在房子的村民数（包括他自己）。狼想知道一开始以及每年改造过后村民的温暖度之和最大是多少。

Input

第一行四个整数 n, m, q, T ，分别表示村民数，房子种类数，年数以及狼改造的种类。

接下来 m 行，每行两个整数，表示 a_i, b_i 。

接下来 q 行：

若 $T = 0$ ，表示狼会拆除房子，则每行一个正整数 x ，表示狼拆除了第 x 种房屋，第 x 种的含义是按照读入顺序的第 x 个。

若 $T = 1$ ，表示狼会额外建造房子，则每行两个正整数 a, b ，表示狼建造了 a 个房屋，每个房屋能容纳 b 个村民。

Output

第一行一个整数表示初始答案。

接下来 Q 行，第 $i + 1$ 行表示第 $i + 1$ 次修改后的答案。。

Example

dynamic.in	dynamic.out
9 0 3 1 4 2 2 3 5 4	9 17 23 33
6 5 3 0 2 3 1 2 3 3 1 5 2 1 4 1 3	26 18 18 8
见选手目录下 dynamic/dynamic3.in	见选手目录下 dynamic/dynamic3.ans

Scoring

对于所有数据, $a_i, b_i \leq 10^6$ 。

测试点编号	n	m	q	T
1~2	≤ 10	≤ 2000	≤ 10	0
3~4	≤ 2000	≤ 10	≤ 10	
5~6	$\leq 10^9$	$\leq 2 \times 10^5$	$= 0$	
7~8	$\leq 10^9$	≤ 2000	≤ 2000	
9~13	$\leq 10^{12}$	$\leq 2 \times 10^5$	$\leq 2 \times 10^5$	
14~15	≤ 10	≤ 2000	≤ 10	1
16~17	≤ 2000	≤ 10	≤ 10	
18	$\leq 10^9$	$\leq 2 \times 10^5$	$= 0$	
19~20	$\leq 10^9$	≤ 2000	≤ 2000	
21~25	$\leq 10^{12}$	$\leq 2 \times 10^5$	$\leq 2 \times 10^5$	

Problem D. 贪心题

Input file: `greedy.in`
Output file: `greedy.out`
Time limit: 2 second
Memory limit: 512 megabytes

小圆过生日，作为小圆忠实粉丝的小 W 想要送给她一个序列 a_i 。

直接送一个序列未免太过直接，所以小 W 要经过一些加工处理：他会选出 k 个**不交**的子区间 $[l_1, r_1], [l_2, r_2], \dots, [l_k, r_k]$ ，其中有 $l_1 \leq r_1 < l_2 \leq r_2 < \dots < l_k \leq r_k$ ，将每个子区间内的物品打包起来。记第 i 个子区间的和为 s_i ，定义这样一个包装方案的美观度为：

$$|s_1 - s_2| + |s_2 - s_3| + \dots + |s_{k-1} - s_k|$$

由于小 W 是小圆的忠实粉丝，所以小 W 想让美观度尽量大，他现在想让你求出这个最大美观度。

注意：最终的包装方案**不一定要**包含序列中的**所有数**，也**不一定是**连续的选择。

Input

第一行两个正整数 n, k 。

接下来一行 n 个整数，描述了这个序列 a_i 。

Output

一行一个数，表示答案。

Example

<code>greedy.in</code>	<code>greedy.out</code>
6 3 -1 2 -2 4 5 1	19
见选手目录下 <code>greedy/greedy2.in</code>	见选手目录下 <code>greedy/greedy2.ans</code>

Notes

对于样例 1：选取区间 $[3, 3], [4, 5], [6, 6]$ ， $s_1 = -2, s_2 = 9, s_3 = 1, |s_1 - s_2| + |s_2 - s_3| = 19$ 。

Scoring

本题采用捆绑测试，只有通过了同一个子任务中的所有数据才能获得该子任务的全部分数。

对于所有数据，有 $2 \leq k \leq n \leq 100000, 2 \leq k \leq 200, |a_i| \leq 10^9$ 。

子任务 1 (5 分)： $n \leq 5$ 。

子任务 2 (15 分)： $n \leq 20$ 。

子任务 3 (20 分)： $n \leq 400, k \leq 50$ 。

子任务 4 (10 分)： $k = 2$ 。

子任务 5 (50 分)：没有特殊限制。