

模拟赛

第一试

题目名称	红黑树	机器人	阿鲁巴
题目类型	传统型	传统型	传统型
目录	rbtree	robot	aruba
可执行文件名	rbtree	robot	aruba
输入文件名	rbtree.in	robot.in	aruba.in
输出文件名	rbtree.out	robot.out	aruba.out
每个测试点时限	4.0 秒	3.0 秒	4.0 秒
内存限制	512 MB	512 MB	512 MB
测试点数目	20	25	25
测试点是否等分	是	是	是

提交源程序文件名

对于 C++ 语言	rbtree.cpp	robot.cpp	aruba.cpp
对于 C 语言	rbtree.c	robot.c	aruba.c
对于 Pascal 语言	rbtree.pas	robot.pas	aruba.pas

编译选项

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

红黑树 (rbtree)

【题目背景】

Yazid 和 cyand1317 是好朋友。

【题目描述】

Yazid 有一棵 n 个节点的有根树，节点的编号从 1 至 n ，其中根节点的编号为 1。

这棵树上的所有节点都可以被染成红色或黑色，因此 Yazid 把它叫做红黑树。

cyand1317 不喜欢黑色，他希望把树上尽可能少的节点染成黑色。

而 Yazid 喜欢黑色，所以他要妨碍 cyand1317。作为这棵树的主人，他给出了一些限制，包含了 a 个 A 类限制和 b 个 B 类限制，每个限制都包含 2 个参数 r, s 。两类限制的意义分别如下：

- A 类限制规定了以 r 为根的子树中，黑色节点的数目不得少于 s 个。
- B 类限制规定了除以 r 为根的子树外，黑色节点的数目不得少于 s 个。

cyand17 对这棵树的染色必须满足 Yazid 的所有限制。

cyand17 绝顶聪明，所以他一定会在满足限制的前提下，染黑尽可能少的节点。你能预测最终黑色节点的数目吗？

【输入格式】

从文件 `rbtree.in` 中读入数据。

本题包含多组数据。第一行一个非负整数 T 表示数据组数，接下来依次描述每组数据。对于每组数据：

第一行一个正整数 n ，表示 Yazid 的红黑树的节点数目。

接下来 $n-1$ 行，每行两个正整数 u, v ，描述一条树上的边。数据保证这些边一定构成了一棵树。

接下来一行一个正整数 a ，表示 A 类限制的数目。

接下来 a 行，每行两个正整数 r, s ，描述一个 A 类限制。

接下来一行一个正整数 b ，表示 B 类限制的数目。

接下来 b 行，每行两个正整数 r, s ，描述一个 B 类限制。

【输出格式】

输出到文件 `rbtree.out` 中。

对于每组数据，输出一行一个整数，表示 cyand1317 染色后的黑点数目，特别地，对于不可能同时满足限制的情况，请输出 -1 。

【样例 1 输入】

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

【样例 1 输出】

```
2
-1
```

【样例 1 解释】

对于第一组数据，cyand1317 只需将节点 2,5 染黑即可。可以证明不存在黑点更少的解。

对于第二组数据，唯一的一个 B 类限制是无法满足的（在以 3 为根的子树外，不可能选出 5 个节点染黑）。

【样例 2】

见选手目录下的 `rbtree/rbtree2.in` 与 `rbtree/rbtree2.ans`。

【子任务】

测试点编号	n	$\sum n$	特殊限制
1~3	≤ 17	≤ 200	无
4~6	$\leq 1,000$	$\leq 10,000$	
7~9	$\leq 10^5$	$\leq 10^6$	$b = 0$
10~12			$b \leq 1$
13~15			$a = 0$
16~18			对于所有树边有 $u = v + 1$
19~20			无

对于一个测试点， $\sum n$ 指的是该测试点中所有数据 n 的总和。

对于所有测试点，保证 $\sum n \leq 10^6$ 。

对于所有测试点的所有测试数据，保证 $n \leq 10^5$ ，保证不存在 r 相同的同类型限制（这也意味着 $a, b \leq n$ ）；对于所有的限制，保证 $0 \leq s \leq n$ 。

机器人 (robot)

【题目背景】

cyand1317 和 wangyurzee7 是好朋友。

【题目描述】

cyand1317 有一条数轴，于是他邀请 wangyurzee7 一起来玩游戏。

cyand1317 和 wangyurzee7 各有一个机器人。0 时刻时，他们都处于数轴的整点上（并不一定处于相同的位置）。

cyand1317 会向他的机器人依次下达 n 条指令（指令编号从 1 开始）。其中第 i 条指令的内容是：在接下来的 A_i 个时刻都以 V_i 每时刻的速度在数轴上**匀速运动**（ V_i 的取值范围为 $\{-1, 0, 1\}$ ）。这意味着，cyand1317 的机器人会在 $[\sum_{j=1}^{i-1} A_j, \sum_{j=1}^i A_j)$ 时刻内，每时刻都向正方向移动 V_i 单位长度。

wangyurzee7 也会类似地向他的机器人依次下达 m 条指令。第 i 条指令的内容是：在接下来的 B_i 个时刻都以 W_i 每时刻的速度在数轴上**匀速运动**（ W_i 的取值范围同样为 $\{-1, 0, 1\}$ ）。

由于 cyand1317 和 wangyurzee7 将会同时结束游戏，所以有 $\sum_{i=1}^n A_i = \sum_{i=1}^m B_i$ ，我们也认为这个时刻是整个游戏的最后一个时刻，我们将它记作 L 。

对于任意的**整数**时刻，如果两个机器人处于坐标轴上相同位置，那么 cyand1317 会认为这个时刻是**优秀的**。

由于游戏时间太长，cyand1317 和 wangyurzee7 早就忘记了他们机器人的初始坐标。不过他们都把自己的所有指令按顺序记录了下来。

wangyurzee7 对优秀的时刻数非常感兴趣。他想知道，对于所有可能的两机器人起始位置情况，最多可能有多少优秀的时刻。

无助的 wangyurzee7 并不能弄清机器人的运行过程，所以请你帮帮他。

【输入格式】

从文件 **robot.in** 中读入数据。

本题包含多组数据。第一行一个非负整数 T 表示数据组数，接下来依次描述每组数据。对于每组数据：

第 1 行一个正整数 n ，表示 cyand1317 的指令数。

第 2 行到第 $n+1$ 行，第 $i+1$ 行两个用空格隔开的整数 V_i, A_i ，描述 cyand1317 的第 i 条指令。

第 $n+2$ 行一个正整数 m ，表示 wangyurzee7 的指令数。

第 $n+3$ 行到第 $n+m+2$ 行，第 $i+n+2$ 行两个用空格隔开的整数 W_i, B_i ，描述 wangyurzee7 的第 i 条指令。

【输出格式】

输出到文件 *robot.out* 中。

对于每组数据，输出一行一个整数表示答案。

【样例 1 输入】

```
2
1
1 2
2
1 1
-1 1
1
0 6
4
-1 2
1 1
-1 2
1 1
```

【样例 1 输出】

```
2
3
```

【样例 1 解释】

对于第一组数据，cyand1317 和 wangyurzee7 的机器人的初始坐标如果均为 0，即会有 2 个优秀的时刻：

时刻 0 两机器人的坐标分别为 0 和 0，是优秀的时刻。

时刻 1 两机器人的坐标分别为 1 和 1，是优秀的时刻。

时刻 2 两机器人的坐标分别为 2 和 0。

可以证明不可能存在更多优秀的时刻。

对于第二组数据，cyand1317 和 wangyurzee7 的机器人的初始坐标如果分别为 2 和 4，即会有 3 个优秀的时刻（时刻 2, 4, 6）。可以证明不可能存在更多优秀的时刻。

【样例 2】

见选手目录下的 *robot/robot2.in* 与 *robot/robot2.ans*。

【子任务】

n, m	$\Sigma(n + m)$	测试点编号	L	V_i	W_i
≤ 100	$\leq 1,000$	1	≤ 200	$\in \{-1, 0, 1\}$	$\in \{-1, 0, 1\}$
$\leq 1,000$	$\leq 10,000$	2	$= n = m$		
		3~4	$\leq 3,000$		
		5~6	$\leq 10^9$		
$\leq 10^5$	$\leq 2 \times 10^6$	7~8	$= n = m$	$\in \{-1, 0, 1\}$	$\in \{-1, 0, 1\}$
		9~10	$\leq 2 \times 10^5$		
		11	$\leq 10^{18}$	$\in \{0\}$	$\in \{-1, 0\}$
		12~14		$\in \{0, 1\}$	
		15~18			
		19~25		$\in \{-1, 0, 1\}$	$\in \{-1, 0, 1\}$

为了方便你阅读，我们把测试点编号放在了表格的中间，请注意。

对于一个测试点， $\Sigma(n + m)$ 指的是该测试点中所有数据 $n + m$ 的总和。

对于所有测试点，保证 $\Sigma(n + m) \leq 2 \times 10^6$ 。

对于所有测试点的所有数据，保证 $1 \leq n, m \leq 10^5$ ， $V_i, W_i \in \{-1, 0, 1\}$ ， $1 \leq A_i, B_i \leq L \leq 10^{18}$ 。

阿鲁巴 (aruba)

【题目背景】

YJQQQAQ 是 Yazid 和 cyand1317 和你的好朋友。

【题目描述】

阿鲁巴是一款风靡全国的游戏。

作为这款游戏狂热的爱好者，Yazid 和 cyand1317 和你打算邀请 YJQQQAQ 体验这款游戏，感受其中的乐趣。

众所周知，进行这款游戏首先需要有一个柱子。

于是，Yazid 搬来了 c 种颜色的 $1 \times 1 \times 1$ 的砖头，每种颜色的砖头都有无限个。

小伙伴们希望用这些砖头造出 Q 个柱子底面为 2×2 正方形的柱子，其中第 i 个柱子的高度不超过 h_i 。也就是说，所有的柱子都是 $2 \times 2 \times x$ 的，对于第 i 个柱子， x 为 $[1, h_i]$ 之间的整数。

严格的 cyand1317 讨厌单调的事物，因此他不希望有超过 k 对相邻的砖头是颜色相同的。我们认为两个砖头是相邻的，当且仅当他们有公共面。

现在 Yazid 和 cyand1317 想知道对于每个柱子，共有多少种造出它的本质不同的方案。需要注意的是，两种可以通过旋转互相得到对方的方案不会被认为是本质相同的。两种方案被认为是本质相同的，当且仅当对于所有 i, j, l ，都有第 l 层第 i 行第 j 列的砖头在两种方案中颜色相同。

YJQQQAQ 已经迫不及待要体验这款游戏了，所以请你帮助 Yazid 和 cyand1317 计算。你只需要告诉他们答案对 998,244,353 取模的结果即可。

【输入格式】

从文件 *aruba.in* 中读入数据。

第一行 2 个用空格隔开的正整数 c, k ，分别表示颜色的数目、最多的相邻同色砖头对数。

第二行一个非负整数 Q ，表示柱子的总数。

第 3 行到第 $Q + 2$ 行，第 $i + 2$ 行一个正整数 h_i ，表示第 i 根柱子的最大高度。

【输出格式】

输出到文件 *aruba.out* 中。

输出 Q 行每行一个整数，第 i 行的整数表示造第 i 根柱子的方案数对 998,244,353 取模的结果。

【样例 1 输入】

```
2 0
3
1
2
3
```

【样例 1 输出】

```
2
4
6
```

【样例 1 解释】

由于不允许存在相邻的同色砖头，所以第一层只有 2 种方案。第一层确定方案后，更高层的方案也随之确定。因此总方案数为底层方案数（即 2）与层数上限的乘积。

【样例 2 输入】

```
2 3
1
1
```

【样例 2 输出】

```
14
```

【样例 3 输入】

```
4 7
5
1
10
47
233
6666
```

【样例 3 输出】

256
378680255
317260259
349223183
155840265

【子任务】

测试点编号	c	k	Q	h_i
1	$= 2$	$= 0$	$= 100$	≤ 200
2	$= 3$			
3	$= 4$			
4	$\leq 1,000$			
5	$= 2$	≤ 7		
6	$= 3$			
7	$= 4$			
8	$\leq 1,000$			
9	$= 2$	$= 0$	$= 200$	$\leq 10^9$
10	$= 3$			
11	$= 4$			
12	$\leq 1,000$			
13	$= 2$	≤ 7	$= 1$	$\leq 10^{18}$
14	$= 3$			
15	$= 4$			
16	$\leq 1,000$			
17	$= 1$			
18	$= 2$			
19	$= 3$			
20	$= 4$			
21	$\leq 1,000$			
22~25	$\leq 10^9$			

对于所有测试点，保证 $1 \leq c \leq 10^9$ ， $0 \leq k \leq 7$ ， $Q \leq 200$ ， $h_i \leq 10^{18}$ 。