预览输出

题目名称	归程
题目类型	传统型
目录	return
可执行文件名	return
输入文件名	return.in
输出文件名	return.out
每个测试点时限	1.0 秒
内存限制	512 MB
测试点数目	0
每个测试点分值	??

提交源程序文件名

对于 C++ 语言	return.cpp
对于 C 语言	return.c
对于 Pascal 语言	return.pas

编译选项

对于 C++ 语言	-O2 -std=c++11
对于 C 语言	-O2 -std=c11
对于 Pascal 语言	-O2

归程（return）

【题目背景】

Captain.W. 是一名来自魔力之都的 OIer，他将踏上归程，回到他温暖的家。

【题目描述】

魔力之都可以抽象成一个 n 个节点、m 条边的**无向连通图**（节点的编号从 1 至 n）。Captain.W. 的家恰好在魔力之都的 n 号节点。

每无向条边有 3 个属性：长度、宽度、海拔。我们分别用 l, w, a 描述这三个属性。它们都是顾名思义的，这里不再赘述。

作为季风气候的代表城市，魔力之都时常有雨水相伴。魔力之都的排水系统非常神奇，每条边的排水系统都在地下相连，然而地下排水系统能容纳水的体积却非常小，达到了可以忽略不计的地步。这也就意味着，**对于所有有积水的边，它们的水位海拔是一致的**。

这里特别需要说明的是，节点的面积是可以忽略不计的，也就是说你可以认为节点不会积水。

Captain.W. 在接下来 Q 天都要从不尽相同的地点回到他的家：

- 每一天，Captain.W. 都会告诉你他的出发点 v ，以及当天的**降水量** p。
  - 降水量表示单位面积**积水高度**。你同样可以理解为所有道路积水高度按面积加权平均。
  - 特别提醒：请注意这里的积水高度与上面提到的水位海拔之间的区别（你也可以辅助样例来理解）。
- 每一天，Captain.W. 在出发点都拥有一辆**车**。这辆车 —---由于某些原因 —---不能经过**有积水**的边。
- Captain.W. 可以在任意节点下车，这样接下来他就可以步行经过**有积水**的边。但车会被留在他下车的节点，他只有步行回到该节点才能重新上车。（在第二天车会被重置）
- Captain.W. 非常讨厌在雨天步行，因此他希望在完成回家这一目标的同时，最小化他**步行经过的**边的总长度。

请你帮助 Captain.W. 进行计算。

本题的部分测试点将强制在线，具体细节请见【输入格式】和【子任务】。

【输入格式】

从文件 ***return.in*** 中读入数据。

第一行 2 个正整数 n, m，分别表示节点数、边数。

接下来 m 行，每行 5 个正整数 u, v, l, w, a，描述一条连接节点 u, v 的、长度为 l、宽度为 w、海拔为 a 的边。

接下来一行 2 个非负数 Q, K, S，其中 Q 表示总天数，$K \in \{0, 1\}$ 是一个会在下面被用到的系数，S 表示的是可能的最大降水量值。

接下来 Q 行依次描述每天的状况。每行 2 个非负整数 v_0, p_0 描述一天：

- 这一天的出发节点为 $v = (v_0 + K \times lastans - 1) \bmod n + 1$。
- 这一天的降水量为 $p = (p_0 + K \times lastans) \bmod (S + 1)$。
- 其中 $lastans$ 表示上一天的答案（最小步行总路程）。特别地，我们规定第 1 天时 $lastans = 0$。

对于输入中的每一行，如果该行包含多个数，则用单个空格将它们隔开。

【输出格式】

输出到文件 ***return.out*** 中。

输出 Q 行每行一个整数，依次表示每天的最小步行总路程。

【样例 1 输入】

```
3 2
1 2 50 15 6
2 3 100 10 13
4 0 10
3 0
2 1
3 3
3 4
```

【样例 1 输出】

```
0
50
50
150
```

【样例 1 解释】

第一天没有降水，Captain.W. 可以坐车直接回到家中。

第二天连接 1,2 号节点的边有积水，因此 Captain.W. 从 2 号点出发坐车只能去往 3 号节点，对回家没有帮助。他只能直接步行回到家中。

第三天这条海拔较低的边仍然积水，但另一条边恰好未积水。因此 Captain.W. 可以坐车先到达 2 号节点，再步行回家。

第四天所有的边都积水了，因此 Captain.W. 只能纯靠徒步回家。

如下五图依次为：魔力之都的俯视图、第 1 至 4 天每天的降水高度图。

咕咕咕咕咕

【样例 2 输入】

```
5 6
1 2 1 1 4
2 3 1 1 4
4 3 1 1 4
5 3 1 1 4
1 4 2 1 1
1 5 2 1 1
4 1 3
5 1
3 0
1 0
5 2
```

【样例 2 输出】

```
2
3
3
2
```

【样例 2 解释】

本组数据强制在线。

第一天的答案是 2，因此第二天的 $v = (3 + 2 - 1) \bmod 5 + 1 = 5$，$p = (0 + 2) \bmod (3 + 1) = 2$。

第二天的答案是 3，因此第三天的 $v = (1 + 3 - 1) \bmod 5 + 1 = 4$，$p = (0 + 3) \bmod (3 + 1) = 3$。

第三天的答案是 3，因此第四天的 $v = (5 + 3 - 1) \bmod 5 + 1 = 3$，$p = (2 + 3) \bmod (3 + 1) = 1$。

【样例 3】

见选手目录下的 ***return/return3.in*** 与 ***return/return3.ans***。

【样例 4】

见选手目录下的 ***return/return4.in*** 与 ***return/return4.ans***。

【子任务】

为了方便你阅读下面的表格，我们对表格中的一些内容作如下说明：

- 图形态：对于表格中该项为 ‘‘一棵树’’ 或 ‘‘一条链’’ 的测试点，保证 $m = n - 1$。除此之外，这两类测试点分别满足如下限制：
  - 一棵树：保证输入的图是一棵树，即保证边不会构成回路。
  - 一条链：保证所有边满足 $u + 1 = v$。
- 海拔：对于表格中该项为 ‘‘两种’’ 或 ‘‘一种’’ 的测试点，分别满足如下性质：
  - 两种：保证对于所有边，有 $a \in \{1, 2\}$。
  - 一种：保证对于所有边，有 $a = 1$
- 强制在线：对于表格中该项为 ‘‘是’’ 的测试点，保证 $K = 1$；如果该项为 ‘‘否’’，则有 $K = 0$。
- 对于所有测试点，如果上述对应项为 ‘‘不保证’’，则对该项内容不作任何保证。

为了优化你的阅读体验，我们把测试点的编号放在了表格的中间，请注意这一点。

<table>
<tr><th>$n=$</th><th>$m=$</th><th>$Q=$</th><th>测试点</th><th>图形态</th><th>海拔</th><th>强制在线</th></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td rowspan="4">不保证</td><td rowspan="4">两种</td><td rowspan="7">否</td></tr>
<tr><td>6</td><td>10</td><td>10</td><td>1</td></tr>
<tr><td>60</td><td>200</td><td>100</td><td>2</td></tr>
<tr><td>100</td><td>300</td><td>200</td><td>3</td></tr>
<tr><td>2000</td><td>1999</td><td rowspan="15">300000</td><td>4</td><td rowspan="3">一条链</td><td rowspan="4">不保证</td></tr>
<tr><td rowspan="3">200000</td><td rowspan="3">199999</td><td>5</td></tr>
<tr><td>6</td><td rowspan="2">是</td></tr>
<tr><td>7</td><td>一棵树</td></tr>
<tr><td>2000</td><td>5000</td><td>8</td><td rowspan="12">不保证</td><td rowspan="3">一种</td><td>否</td></tr>
<tr><td rowspan="11">200000</td><td rowspan="11">400000</td><td>9</td></tr>
<tr><td>10</td><td rowspan="2">是</td></tr>
<tr><td>11</td><td>两种</td></tr>
<tr><td>12</td><td rowspan="10">不保证</td><td rowspan="5">否</td></tr>
<tr><td>13</td></tr>
<tr><td>14</td></tr>
<tr><td>15</td></tr>
<tr><td>16</td><td rowspan="5">是</td></tr>
<tr><td>17</td></tr>
<tr><td>18</td></tr>
<tr><td>19</td></tr>
</table>

所有测试点均保证满足如下限制：

- $n \le 2 \times 10^5$，$m \le 4 \times 10^5$，$Q \le 3 \times 10^5$，$K \in \{0, 1\}$，$1 \le S \le 10^5$。
- 对于所有边：$1 \le u, v \le n$，$l \le 10^4$，$w \le 10^4$，$a \le 2 \times 10^5$。
- 任意两点之间都直接或间接通过边相连。
- 对于所有询问：$1 \le v_0 \le n$，$p_0 \le S$。

【提示】

这里是一个非常温馨的提示。