

士兵 (soldier)

【题目描述】

有 n 个士兵站成一排。初始时，编号为 i 的士兵血量为 a_i 。

你可以进行若干次（也可以不进行）进攻。每次进攻时，你可以选择一段区间 $[l, r]$ ($1 \leq l \leq r \leq n$)，将编号在这一段区间内的士兵的血量减少 1。每次进攻需要付出 m 的代价。

在所有的进攻结束后，若第 i 个士兵的血量不超过 0，那么你会获得 b_i 的收益。由于士兵中敌友混杂， b_i 有可能是负数。注意每个士兵的收益只会获得一次。

你需要找到一个进攻方案，来最大化收益总和减去代价总和。形式化地，设你的方案进攻了 k 次，最终第 i 个士兵的血量为 c_i ，令 p_i 在 $c_i \leq 0$ 时为 1 否则为 0，则需要最大化 $(\sum_{i=1}^n p_i \times b_i) - m \times k$ 。最终，你需要给出收益总和减去代价总和的最大值即可。

【输入格式】

从文件 *soldier.in* 中读入数据。

本题有多组测试数据。

输入的第一行包含一个正整数 T 表示数据组数。

接下来依次输入每组测试数据，对于每组测试数据：

第一行包含两个整数 n, m ，分别表示士兵数量和单次进攻代价。

接下来 n 行，每行包含两个整数 a_i, b_i ，分别描述每个士兵的血量和收益。

【输出格式】

输出到文件 *soldier.out* 中。

对于每组数据输出一行包含一个整数，表示在所有可能的方案中可以得到的收益总和减去代价总和的最大值。

【样例 1 输入】

```
1 3
2 5 1
3 1 3
4 2 5
5 1 4
6 3 3
7 5 1
```

```

8 3 2
9 1 5
10 1 -100
11 1 5
12 3 2
13 1 5
14 1 -1
15 1 5

```

【样例 1 输出】

```

1 12
2 6
3 7

```

【样例 1 解释】

第一组测试数据中，一个最优方案为进攻三次，分别取区间 $[1, 5], [2, 4], [4, 4]$ ，最终血量序列为 $(0, 0, -1, 0, 4)$ ，收益为 $b_1 + b_2 + b_3 + b_4 = 15$ ，代价为 $3 \times 1 = 3$ 。可以证明没有比该方案更优的方案，故输出 $15 - 3 = 12$ 。

第二组测试数据中，一个最优方案为进攻两次，分别取区间 $[1, 1], [3, 3]$ 。

第三组测试数据中，一个最优方案为进攻一次，取区间 $[1, 3]$ 。

【子任务】

设 $\sum n$ 表示单个测试点中所有测试数据的 n 的和。对于所有测试数据，保证

- $1 \leq T \leq 5 \times 10^5$,
- $1 \leq n, \sum n \leq 5 \times 10^5, 1 \leq m \leq 10^9$,
- $\forall 1 \leq i \leq n, 1 \leq a_i \leq 10^9, -10^9 \leq b_i \leq 10^9$ 。

子任务编号	分值	$\sum n \leq$	特殊性质
1	5	5×10^5	$\forall 1 \leq i \leq n, b_i \geq 0$
2	15	5,000	$\forall 1 \leq i \leq n, a_i \leq 5,000$
3	20		
4	25	10^5	无
5	25	2×10^5	
6	15	5×10^5	