

注意, 针对所有题目, 请给出算法的复杂度说明.

A. 任务分配

公司每天会有很多新的任务. 在接下来的 n 天, 有 m 个任务需要处理. 各个任务都有截至日期 d_i , 任务收益为 g_i , 如果没有在截至日期 d_i 之前完成则无法获得收益, 但也不会有损失. 在第 d_i 天完成任务也可以得到收益.

公司现在有 r 名员工. 最近大家都想放年假, 所以每个人只想就从第 l_i 天工作到第 r_i 天(包括第 r_i 天), 然后就出去玩了. 每个人处理任务的能力也不同, 处理能力为 w_i , 表示在这几天的工作中一共能处理 w_i 项任务量. 现在想让你帮我们分配一下任务, 使得接下来的 n 天总利润最大.

输入格式:

第一行输入 n, m, r . 保证 $1 \leq n, m, r \leq 2 \times 10^5$.

接下来 m 行, 每行有2个值 d_i, g_i . 保证 $1 \leq d_i \leq n, g_i > 0$.

接下来 r 行, 每行有3个值 l_i, r_i, w_i . 保证 $1 \leq l_i \leq r_i \leq n, w_i > 0$.

输出格式:

输出 v , 表示最大总利润.

示例1:

输入

```
3 5 2
2 3
1 10
2 10
3 2
3 10
1 3 2
1 2 2
```

输出

```
33
```

说明

1号员工从第1天工作到第3天, 完成第1项和第5项工作, 利润为13;
2号员工从第1天工作到第2天, 完成第2项和第3项工作, 利润为20;
则总利润为33.
方案不唯一, 输出最大总利润即可.

B. 超级计算机

现在超级计算机SSoM需要顺序执行 n 个任务. 这些任务有读写和计算两个部分, 第 i 个任务需要读写 r_i 个单位的数据, 并计算 c_i 个单位的数据. SSoM有两种状态:计算状态和读写状态. 如果SSoM的处理能力为 t , 在计算状态下, 处理第 i 个任务耗时为 $\lceil \frac{r_i}{2t} \rceil + \lceil \frac{c_i}{t} \rceil$;在读写状态下,处理第 i 个任务耗时为 $\lceil \frac{r_i}{t} \rceil + \lceil \frac{c_i}{2t} \rceil$. 因为频繁切换会减少机器寿命, 所以最多只能切换 m 次. SSoM有多个型号, 每个型号的处理能力不同. 为找到适合的型号处理任务, 现在想知道 t 最小为多大的正整数才能在时间 T 之内完成所有任务. SSoM初始为计算状态.

输入格式:

第一行输入 $n \leq 10^3, m \leq 10^3, T \leq 10^6$;

接下来 n 行, 每行输入 $r_i \geq 0, c_i \geq 0$.

输入保证有解.

输出格式:

输出整数 t ,表示完成任务需要的最小处理能力.

示例1:

输入

```
6 2 20
1 1
2 3
2 2
4 3
1 4
7 2
```

输出

```
2
```

C. mrank

函数mrank在线地接收一个数字, 并返回其在所有已接收的数字中的排序. 每个元素的排序仅仅与其之前的元素相关, 与其之后的元素无关.

示例1:

输入

```
mrank(1)
mrank(3)
mrank(2)
mrank(3)
mrank(4)
```

输出

0
1
1
2
4