子图

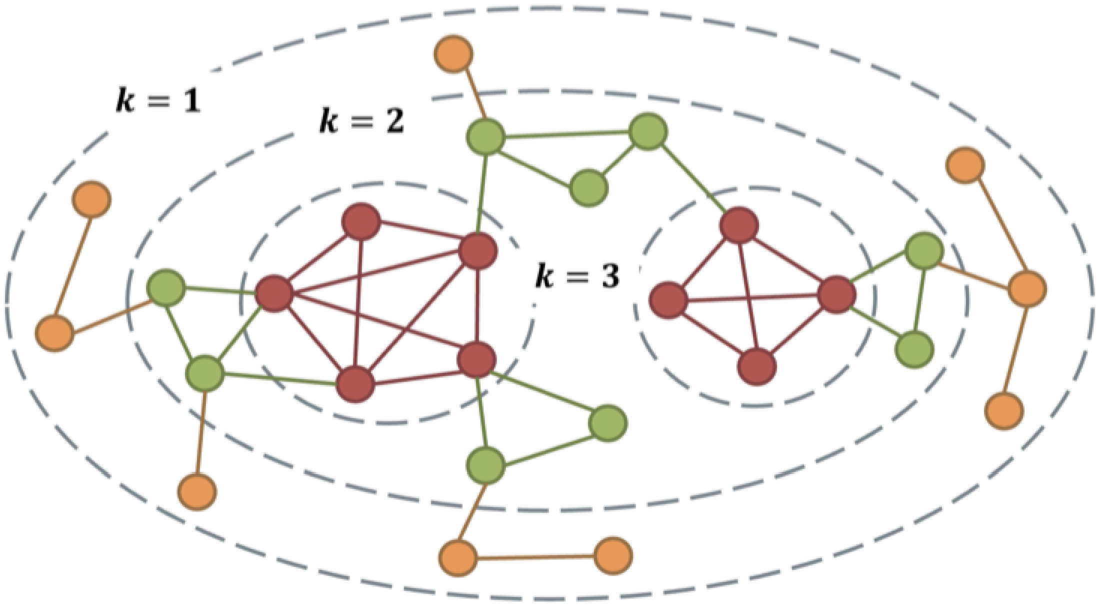
时间限制：1000ms 空间限制：512MB d.in/out

问题描述

Cuber QQ 的研究兴趣是在图 $G=(V(G),E(G))$ 中找到最好的 k-degree 子图。

子图 S 是 G 的 k-degree 子图需要满足以下要求：

- 每个顶点 $v(v \in S)$ 在 S 中至少有 k 度；
- S 是连通的；
- S 是极大的，即 S 的任何超图都不是 k-degree 子图，除了 S 本身。



然后 Cuber QQ 定义子图 S 的分数。在定义分数之前，他首先定义：

- $n(S)$：子图 S 中的顶点数，即 $n(S)=|V(S)|$；
- $m(S)$：子图 S 的边数，即$m(S)=|E(S)|$；
- $b(S)$：子图 S 中的边界边数，
  $b(S)=|\{(u,v)|(u,v)\in E(G),u\in V(S),v\notin V(S),v\in V(G)\}|$;

他定义一个子图的分数为 $score(S)=M\cdot m(S)-N\cdot n(S)+B\cdot b(S)$，其中 M,N,B 是给定的常数。

子图的分数越高，Cuber QQ 认为它越好。你需要在图 G 中找到最好的 k-degree 子图。如果有许多 k-degree 子图的分数相同，则应最大化 k。

输入格式

第一行包含两个整数 n 和 m ($1\le n,m\le 10^6$)，其中 n 和 m 是图中的点数和边数。

第二行包含三个整数 M,N 和 B ($-10^9\le M,N,B\le 10^9$)，其中一个子图的得分为 $score(S)=M\cdot m(S)-N\cdot n(S)+B\cdot b(S)$。

接下来的每一行 m 包含两个整数 u 和 v（$1\le u,v\le n,u\ne v$），表示一条边。

保证给定的图没有自环和重边且给定的图一定是无向图。

输出格式

一行，包含两个空格分隔的整数，它们是 k 和最佳 k-degree 子图的分数。

应该确保 $k > 0$ 。

样例

输入

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

输出

```
2 0
```

数据范围

对于其中 30% 的数据，保证 $1 \le n, m \le 30$；

对于其中 50% 的数据，保证 $1 \le n, m \le 1000$；

对于其中 70% 的数据，保证 $1 \le n, m \le 10^5$。