

还款

[问题描述]

A 欠 B 钱，共 N 元 ($1 \leq N \leq 10^{12}$)。A 必须在 K 天内将钱给 B。但是，A 不想将钱太早拿出手。另一方面，A 不得不在还债上有所进展，所以 A 必须每天给 B 至少 M 元 ($1 \leq M \leq 10^{12}$)。

以下是 A 决定偿还 B 的方式。首先 A 选择一个正整数 X 。然后 A 每天都重复以下过程：

- 1、假设 A 已经给了 B 共 G 元，计算 $(N-G)/X$ 向下取整。令这个数为 Y 。
- 2、如果 Y 小于 M ，令 Y 等于 M 。
- 3、给 B 钱， Y 元。

求 X 的最大值，使得 A 按照上述过程能够在 K 天后给 B 至少共 N 元 ($1 \leq K \leq 10^{12}$)。

[输入格式]

一行，包含三个空格分隔的正整数 N 、 K 和 M ，满足 $K \times M < N$ 。

[输出格式]

输出最大的正整数 X ，使得按照上述过程 A 会给 B 至少 N 元。

[输入样例]

```
10 3 3
```

[输出样例]

```
2
```

[说明] 在这个测试用例中，当 $X=2$ 时，A 第一天给 B 钱为 5 元，后两天每天给 B 的钱为 $M=3$ 元。

[注意] 这个问题涉及到的整数规模需要使用 64 位整数类型（例如，C/C++ 中的“long long”）。

[测试点性质] 测试点 2~4 满足 $K \leq 10^5$ ；测试点 5~11 没有额外限制。