

Day 7

——“我是不是在哪见过你？”

计算器 (calc.in/out, 2s, 256MB)

【题目描述】

Maxim 得到了一个计算器，这个计算器有两个整数单元，一开始，第一个单元包含数字 1，第二个单元包含数字 0。这个计算器支持以下两种操作：

1. 假设第一个单元的数字为 a ，第二个单元的数字为 b ，那么将第二个单元的数字改成 $b+1$ 。

2. 假设第一个单元的数字为 a ，第二个单元的数字为 b ，那么将第一个单元的数字改成 $a*b$ 。

现在 Maxim 想知道，有多少个正整数 $x(l \leq x \leq r)$ 满足，存在一种方式从计算器初始状态开始，操作不超过 p 步之后使得第一个单元中的数字为 x 。

【输入格式】

共一行，为三个空格隔开的正整数 l 、 r 、 p

【输出格式】

输出仅一行一个整数 ans ，即为问题所求。

【样例输入】

2 111 11

【样例输出】

47

【数据范围】

30%的数据满足， $2 \leq l \leq r \leq 10^6, 1 \leq p \leq 15$

100%的数据满足， $2 \leq l \leq r \leq 10^9, 1 \leq p \leq 100$

匹配（pat.in/out, 1s, 256MB）

【题目描述】

给出一个长度为 n 的字符串 A 和一个长度为 m 的模式串 B 。

求 A 中哪些位置的子串和 B 匹配。

匹配时只要串内部大小关系相同就可以。比如“5 6 7”、“1 9 233”和“1 2 3”都是互相匹配的，“2 3 3 9 1”和“3 5 5 6 2”是匹配的。

【输入格式】

第一行三个数 n 、 m 、 p ，其中 p 表示字符串中的字符是 $1 \sim p$ 中的数字。

接下来 n 行表示 A ，接下来 m 行表示 B 。

【输出样例】

第一行输出一个整数 B 。接下来 B 行，每行一个整数，从小到大表示一个子串的起始位置（位置从 1 开始编号）。

【样例输入】

```
5 2 10
1
2
3
4
5
6
7
```

【样例输出】

```
4
1
2
3
4
```

【数据范围】

30%的数据满足： $1 \leq n \leq 100$ ， $1 \leq m \leq 25$ ， $1 \leq p \leq 25$

100%的数据满足： $1 \leq n \leq 100,000$ ， $1 \leq m \leq 25,000$ ， $1 \leq p \leq 25$

最长上升子序列 (array.in/out, 1s, 256MB)

【题目描述】

给出一个长度为 n 的数列 B ，数列中的数都是不超过 m 的正整数。将数列 B 复制 t 次拼接起来，形成一个长度为 $n \cdot t$ 的数列 A 。求 A 的最长上升子序列。

【输入格式】

第一行 n 、 m 、 t
第二行 n 个数表示 B

【输出格式】

一个数，最长上升子序列长度

【样例输入】

3 5 2
3 2 1

【样例输出】

2

【数据范围】

30%的数据， $n \cdot t \leq 3 \cdot 10^5$
100%的数据， $1 \leq n, m \leq 10^5$ ， $1 \leq t \leq 10^9$ ， $n \cdot m \leq 5 \cdot 10^7$