

## Problem A. Yasaka with Postfix Expression

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

所谓后缀表达式是指这样的一个表达式：式中不再引用括号，运算符号放在两个运算对象之后，所有计算按运算符号出现的顺序，严格地由左而右新进行（不用考虑运算符的优先级）。

如： $3 * (5 - 2) + 7$ 对应的后缀表达式为：3.5.2.-\*7.+@。 '@'为表达式的结束符号。 '.'为操作数的结束符号。

### Input

输入：后缀表达式

### Output

输出：表达式的值

### Sample

standard input	standard output
2.3.5.-7.*@	-14

## Problem B. Yasaka with Physical Examination

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

一年一度的学校体检又来了。

考虑到体检室空间有限，每次进入体检室的只能有 $k$ 名同学，然后站成一排。

已知参加体检的学生一共有 $n$ 种不同的体重。

现在问你，在 $k$ 个位置构成一排的情况下，相同体重相邻的情况有多少种？

这里假设学生人数无限（答案可能很大，你只需输出最后的结果 $\text{mod}10^9$ 即可）

### Input

输入两个正整数( $1 \leq k, n \leq 1000$ )，中间用空格隔开

### Output

输出答案对 $10^9$ 求余的结果

### Sample

standard input	standard output
3 2	6

### Note

假设两种情况的体重为 $A$ 和 $B$ ，则存在以下6种情况组合：

(1) $AAB$  (2) $AAA$  (3) $BAA$  (4) $BBA$  (5) $BBB$  (6) $ABB$

## Problem C. Yasaka with Prime

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

Enter a positive integer  $n$  and find the difference between the last prime and the previous one of  $n$ .

If  $n$  is a prime, output 0.

Here the input integer  $n$  does not exceed 1299709 (the 100000<sup>th</sup> prime).

For example: input  $n = 27$ , output  $6(29 - 23)$ .

### Input

Enter a positive integer  $n(1 \leq n \leq 1299709)$

### Output

Output the answer

### Sample

standard input	standard output
26	6

## Problem D. Yasaka with WF

Input file: standard input

Time limit: 3 seconds

Output file: standard output

Memory limit: 512 megabytes

Yasaka去了WF手办展，展上共出售 $n$ 种手办，第 $i$ 种手办价格为 $p_i$ 元，Yasaka有 $m$ 元的预算，想购买尽量多的手办回来，当然Yasaka不想重复购买同一种手办。

WF前期通过各种渠道赠送了全场通兑的优惠券，第 $i$ 种手办如果使用一张优惠券，可以以 $c_i$ 元购买原价 $p_i$ 元的该手办，精打细算的Yasaka共搜集了 $k$ 张优惠券。

展会时间紧迫，请你帮Yasaka决定如何使用这些优惠券可以买到最多的手办。

### Input

第一行为 $n, m, k$  ( $1 \leq n \leq 10^5, 1 \leq m \leq 10^{12}, 1 \leq k \leq n$ )

接下来 $n$ 行中的每一行包含两个整数 $p_i$ 与 $c_i$ ，表示一个原价为 $p_i$ 且优惠价格为 $c_i$ 的手办 ( $1 \leq c_i \leq p_i \leq 10^9$ )

### Output

输出Yasaka最多能买回多少手办

### Sample

standard input	standard output
4 7 1 3 2 2 2 8 1 4 3	3

### Note

对于样例，可以将优惠券用在第三种手办上，再以原价购买第一、二种，共花费6元

## Problem E. Yasaka with Kanako

Input file: standard input

Time limit: 5 seconds

Output file: standard output

Memory limit: 512 megabytes

守矢神社门前的十字路口发生了严重的时空干涉，起因是东西方向将有 $n$ 辆车在某些时刻 $x_1, x_2, \dots, x_n$ 驶过该十字路口，南北方向将有 $m$ 辆车在某些时刻 $y_1, y_2, \dots, y_m$ 驶过该十字路口。



控制时间的流动对神明来说可是小菜一碟，将时间拨回一切发生前，Yasaka Kanako试着让南北方向上的 $m$ 辆车整体停下等待若干个整数时刻再驶入路口，以阻止一切的发生。

节约神力起见，停止的秒数越少越好。

### Input

第一行为 $n$ 和 $m$  ( $1 \leq n, m \leq 2000$ )

第二行为 $x_1, x_2, \dots, x_n$  ( $1 \leq x_i \leq 10^9$ )

第三行为 $y_1, y_2, \dots, y_m$  ( $1 \leq y_i \leq 10^9$ )

数据保证 $x_1 < x_2 < \dots < x_n$ 且 $y_1 < y_2 < \dots < y_m$

### Output

一个非负整数，表示南北方向上的车最少需要整体等待的秒数

### Samples

standard input	standard output
2 1 1 3 2	0
3 3 1 2 7 1 3 5	3

## Problem F. Yasaka with fAKe Tree

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

fAKe Tree是一棵对自己要求很高的生成树

它给自己定义了一个根，节点编号为1

它给自己的每个节点定义了该点深度为它到根的简单路径上经过的点数(包含两端点，根深度为1)

它给自己的每个节点定义了 $a_i$ 代表一条限制：在 $i$ 号节点没有被访问前，不能访问其他深度 $> a_i$ 的点

现在你需要访问这棵树上的每一个点一次且恰好一次，请你给出字典序最小的访问序列

你可以在任意时刻访问任意节点，除了序列 $a$ 对深度的限制以外，没有其他限制

### Input

第一行为节点数 $n(1 \leq n \leq 2 \times 10^5)$

第二行为 $a_1, a_2, \dots, a_n$ (点 $i$ 的深度 $\leq a_i \leq n$ )

第3 ~  $n + 1$ 行共 $n - 1$ 行，每行两个数 $x$ 与 $y$ ，表示树上的边，数据保证构成一棵树

### Output

输出 $n$ 行，每行一个节点编号，表示字典序最小的访问序列

数据保证至少有一种合法访问序列

### Sample

standard input	standard output
5	1
1 5 3 3 2	4
1 5	5
5 3	2
1 4	3
4 2	

### Note

两个访问序列 $c_1, c_2, \dots, c_n$ 与 $d_1, d_2, \dots, d_n$ ， $c$ 比 $d$ 字典序小，当且仅当存在一个下标 $j$ ，使得 $c_j < d_j$ 且对于所有 $i(1 \leq i < j)$ ， $c_i = d_i$

## Problem G. Yasaka with 星間飛行

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

(<·ω·)☆KIRA

终于，你有属于自己的飞船啦！你缜密地观察着帝国的地图，你发现帝国由  $n$  个星球， $m$  条航道组成，形成了一张没有自环，没有重边的无向连通图。现在，你在你的家园 *Frontier*，也就是你眼中的 1 号星球，你想要前往  $n$  号星球看李兰卡的演唱会。

你的飞船有两项属性：

- 最大折越距离  $D$ ：你无法在任何长度大于  $D$  的航道上进行折越。
- 最大折越次数  $E$ ：你最多只能折越  $E$  次。

很遗憾的是，你的飞船的这两项属性在一开始都为 0。但别灰心！你可以花费  $c$  个金币使你的飞船升 1 级，这可以让你的  $D$  和  $E$  同时增加，其中  $D$  增加了  $d$ ， $E$  增加了  $e$ 。

现在你想知道，你最少需要花费多少个金币才能到达  $n$  号星球呢。

### Input

第一行两个正整数  $n, m (2 \leq n \leq 10^5, n-1 \leq m \leq 10^5)$ 。

第二行三个正整数  $c, d, e (1 \leq c, d, e \leq 10^4)$ ，含义见题面。

接下来  $m$  行，每行三个正整数  $u, v, w (1 \leq u, v \leq n, 1 \leq w \leq 10^4)$ ，表示星球  $u$  和星球  $v$  之间有一条长为  $w$  的航道。

### Output

输出一个整数表示最小花费，如果无解则输出  $-1$ 。

### Sample

standard input	standard output
5 7 1 1 1 1 2 1 1 3 5 1 4 1 2 3 2 2 4 5 3 4 3 3 5 5	5

## Problem H. Yasaka with Aimo

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

最近，你发现自己太 *Naive* 了，所以你选择了念诗为了提高自己的知识水平。一首诗的知识水平由每句诗句的最长公共前缀决定，具体来说，假设最长公共前缀的长度是  $k$ ，且这首诗由  $m$  句诗句所组成，那么这首诗的知识水平将高达  $k * m$ 。

你是个带诗人，希望从一组诗句当中选择一些组成新的一首诗，使得它的知识水平最高。好在诗句都是由 01 字符串构成的，你可以很轻松的处理它们，这是坠吼的。

### Input

第一行一个正整数  $n(1 \leq n \leq 10^4)$ ，表示接下来有  $n$  句诗句。

接下来  $n$  行，每行一个只包含 0,1 的字符串  $s(1 \leq |s| \leq 200)$ 。

### Output

你只需要输出能组成的最高的知识水平是多少就行了。

### Samples

standard input	standard output
4 0000 0001 10101 010	6
2 010100101010101010 110100101010101010	20



## Problem I. Sayaka with Infinity No.7

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

*Frontier* 是你的家园，在她上面有  $n$  座村庄，村庄间由  $m$  条双向道路所连接。每一条道路对车辆有载重要求。现在有  $q$  辆货车(他们是互相独立的)在运输货物，你想知道，在任何车辆都不超重的情况下，一次最多能运多少货物。

### Input

第一行两个正整数  $n, m (1 < n \leq 10^4, 1 < m \leq 5 \times 10^4)$ 。

接下来有  $m$  行，每行三个正整数  $x, y, z (1 \leq x, y \leq n, x \neq y, 0 \leq z \leq 10^5)$ ，表示  $x$  村与  $y$  村间有一条限重为  $z$  的双向道路(即载重严格大于  $z$  的车辆不能通过)。

接下来一行一个正整数  $q (1 \leq q \leq 3 \times 10^4)$ 。

接下来有  $q$  行，每行两个正整数  $x, y (1 \leq x, y \leq n, x \neq y)$ ，表示这辆货车想从  $x$  村开往  $y$  村。

村庄间可以不联通，可以有重边，但不会有自环。

### Output

共  $q$  行，表示对于某一辆货车，它所能够的最大载重是多少。如果货车不能到达目的地则输出  $-1$ 。

### Sample

standard input	standard output
4 3	3
1 2 4	-1
2 3 3	3
3 1 1	
3	
1 3	
1 4	
1 3	

## Problem J. Kaguya with QAA

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

Kaguya喜欢QAA

给出一串仅包含大写字母的串 $S$ ，长度为 $N$ ，求 $S$ 中不含相邻字符且长度为3的"QAA"子序列个数。

即：

设字符串 $S$ 的第 $i$ 个字符为 $S_i$ ，求满足下列条件的 $\langle x, y, z \rangle$ 三元组个数

$$1 \leq x, y, z \leq N$$

$$x < y - 1$$

$$y < z - 1$$

$$S_x = 'Q', S_y = 'A', S_z = 'Q'$$

### Input

输入仅一行一个字符串 $S$ ，字符串长度 $N$ 满足 $(1 \leq 5000)$

### Output

输出一个整数，即满足条件的三元组的个数

### Samples

standard input	standard output
QQAQQ	1
QAQAQ	0

## Problem K. Kaguya with arithmetic progression

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

Kaguya特别喜欢等差数列。尤其是公差为 1 且全为正整数的等差数列。

显然，对于每一个数  $s$ ，都能找到一个对应的公差为 1 且全为正整数的等差数列各项之和为  $s$ 。

这时，Kaguya想知道，满足这样条件的等差数列，最小的首项是多少。

### Input

输入仅包含一行一个整数  $s(1 \leq s \leq 10^{12})$

### Output

输出两个正整数，分别表示这个等差数列的首项和末项。请注意输出最小的首项。

### Samples

standard input	standard output
16	16 16
1000000007	500000003 500000004

## Problem L. Kaguya with poetry

Input file: standard input

Time limit: 1 second

Output file: standard output

Memory limit: 256 megabytes

Kaguya写了一首 $N$ 行诗句的诗歌，美中不足的是这些诗句并不押韵，黄药师非常想遵循古诗的押韵。诗歌被分为若干段，每段都是四行诗。每一句诗都有一个韵脚，假如A和B表示两种不同的韵脚，每段四行诗的韵脚只可能是 "AABB" "ABAB" "ABBA"和"AAAA"中的一种。

黄药师将诗句的韵脚都编了号，具有相同编号的句子代表有相同的韵脚。现在，黄药师想删掉一些句子，使得剩下的都是遵循押韵规则的四行诗，而且不允许改变诗句的顺序。

现在就问你：如何找出满足条件最长的诗歌？

### Input

第1行为一个整数 $N$  ( $1 \leq N \leq 4000$ )，代表Kaguya写的诗歌的句子数。

第2行为 $N$ 个整数，分别表示每一行诗的韵脚，这些数字都是不超过 $10^9$ 的正数，每个数之间用1个空格隔开。

### Output

一行一个整数 $k$ ，为Kaguya最多能够得到的四行诗个数

### Sample

standard input	standard output
15 1 2 3 1 2 1 2 3 3 2 1 1 3 2 2	3

### Note

提示：

对于样例，删除第3、6、13句诗，就可以分成3个四行诗，分别为：

1 2 4 5

7 8 9 10

11 12 14 15