

## Problem A. 托米的字符串

Input file:           standard input  
Output file:          standard output  
Time limit:          1 second  
Memory limit:        256 megabytes

托米有一个字符串，他经常拿出来玩。这天在英语课上，他学习了元音字母 $a, e, i, o, u$ 以及半元音 $y$ 。"这些字母是非常重要的！"，托米这样想着，"那么我如果随机取一个子串，里面元音占比期望会有多大呢？"

于是，请你求出对于托米的字符串，随机取一个子串，元音( $a, e, i, o, u, y$ )字母占子串长度比的期望是多少。

### Input

读入一个长度不超过 $10^6$ 的只包含小写字母的字符串，即托米的字符串。

### Output

输出所求的期望值，要求相对(绝对)误差不超过 $10^{-6}$ 。

### Example

standard input	standard output
legilimens	0.446746032

## Problem B. 萨博的方程式

Input file: standard input

Output file: standard output

Time limit: 1 second

Memory limit: 256 megabytes

萨博有个方程式:

$$x_1 \text{ xor } x_2 \text{ xor } x_3 \text{ xor } \dots \text{ xor } x_n = k$$

其中 $\text{ xor }$ 指代位运算中的异或符号。

萨博同时还对每个未知数限制了范围为 $0 \leq x_i \leq m_i$ , 希望你计算出解的个数, 最终答案对 $10^9 + 7$ 取模后输出。

### Input

本题设有多组数据, 请处理到文件尾, 保证数据组数不超过100。

对于每组测试数据:

第一行读入2个正整数 $n, k (n \leq 50, k < 2^{31})$ ;

第二行读入 $n$ 个非负整数 $m_1, m_2, m_3, \dots, m_n (m_i < 2^{31})$ 。

### Output

对于每组测试数据, 输出一个数, 为题目所求。

### Example

standard input	standard output
7 127	1
64 32 16 8 4 2 1	0
6 127	6
64 32 16 8 4 2	
4 5	
1 2 3 4	

## Problem C. 纳新一百的石子游戏

Input file:           standard input  
Output file:          standard output  
Time limit:           1 second  
Memory limit:        256 megabytes

纳新一百和乱得尬得在玩取石子的游戏。他们一共有 $N$ 堆石子，第 $i$ 堆有 $a_i$ 颗石子（若 $a_i = 0$ 则表示这是一堆空石子堆）。

纳新一百和乱得尬得轮流进行游戏，纳新一百先手。轮到某个人时，他需要选择一堆非空的石子堆，并拿走任意数量的石子。如果不存在一堆非空的石子堆，则轮到的人输掉游戏。纳新一百想要知道，他的第一轮操作有多少种不同的取法能够保证他最后取得游戏的胜利。假设两个人都是用最优策略在玩游戏，两种操作方式视为不同当且仅当两种方式选取的石子堆的序号不同或取走的石子数量不同。

为了增加趣味性，纳新一百和乱得尬得决定对前 $i$ 堆石子都玩一次游戏，两次游戏相互独立，也就是说，每开始一个新的游戏，石子堆都会被复原。

现在，纳新一百想要知道每一次游戏中，他能够取得胜利的第一轮操作方案数。

### Input

第一行一个正整数 $N$  ( $1 \leq N \leq 10^5$ )，表示石子堆数。第二行 $N$ 个数，表示 $a_i$  ( $0 \leq a_i < 2^{60}$ )。

### Output

$N$ 行，每行一个整数。第 $i$ 行的整数表示用前 $i$ 堆石子玩游戏时，纳新一百在第一轮有多少种操作方式保证自己能获得胜利。如果纳新一百无论如何都不可能赢得游戏，输出0。

### Example

standard input	standard output
6	0
0 0 2 8 7 0	0
	1
	1
	1
	1

## Problem D. 卡拉巴什的字符串

Input file:           standard input  
Output file:         standard output  
Time limit:          4 seconds  
Memory limit:       256 megabytes

卡拉巴什是字符串大师，这天他闲着无聊，又造了个字符串问题。

给定一个长度为 $N$ 字符串 $S$ ，定义后缀 $i$ 为从第 $i$ 个位置开始的后缀，即 $s_i s_{i+1} \dots s_n$ ，定义 $lcp(i, j)$ 为后缀 $i$ 和后缀 $j$ 的最长公共前缀。

卡拉巴什想要知道，每次他给出一组 $i, j$ ，你能否快速告诉他 $lcp(i, j)$ 。

卡拉巴什的好朋友葫芦是字符串宗师，他认为这个题太无聊，于是他想了另一个问题，假设有一个集合 $\{lcp(i, j) | 1 \leq i < j \leq N\}$ ，他想知道这个集合的 $MEX$ 值是多少。一个集合的 $MEX$ 值为最小的没有出现在集合中的非负整数。

这个问题对卡拉巴什来说太容易了，于是葫芦想知道，对于字符串的每一个前缀，对应的集合的 $MEX$ 值是多少。

### Input

第一行一个整数 $T(1 \leq T \leq 10^5)$ ，表示数据组数。

对于每组数据，共一行，表示字符串 $S(1 \leq |S| \leq 10^6)$ 。

输入数据保证所有字符串长度和不超过 $5 * 10^6$ 。保证字符串只包含小写字母。

### Output

对于每组数据，输出 $|S|$ 个整数，表示每一个前缀对应的 $MEX$ 值。

### Example

standard input	standard output
2	0 1 2 3 4
ababa	0 1 2
baa	

## Problem E. 阔力梯的树

Input file:           standard input  
Output file:          standard output  
Time limit:           1 second  
Memory limit:        256 megabytes

阔力梯有一棵树，这棵树有 $N$ 个节点，每个节点按顺序编号为 $1 \sim N$ ，其中，1号节点是根结点。

定义树上一个节点的“结实程度”为，将这个节点的子树中的所有的节点编号拿出来之后，按照从小到大的顺序排列，然后将相邻元素做差之后求平方和。即假设子树的节点编号排序后的序列为 $a_1, a_2, a_3, \dots, a_k$ ，这个节点的“结实程度”就是：

$$\sum_{i=1}^{k-1} (a_{i+1} - a_i)^2$$

现在，阔力梯想要加固这棵树，但是他的资源有限，不能加固所有的节点，所以他想知道每个节点的“结实程度”是多少。

### Input

第一行一个整数 $N$  ( $1 \leq N \leq 10^5$ )，表示节点数。

接下来第2到 $N$ 行，第 $i$ 行一个整数 $p_i$  ( $1 \leq p_i < i$ )，表示节点 $i$ 的父亲。

### Output

$N$ 行，第 $i$ 行一个整数，代表节点 $i$ 的“结实程度”。

### Examples

standard input	standard output
5	4
1	0
1	0
1	0
1	0
5	4
1	3
2	2
3	1
4	0

## Problem F. 采蘑菇的克拉莉丝

Input file: standard input

Output file: standard output

Time limit: 5 seconds

Memory limit: 256 megabytes

克拉莉丝在玩一个采蘑菇的游戏。游戏地图是一张 $N$ 个节点， $N - 1$ 条边的连通无向图。一开始起点在1号点。

游戏过程中会发生两种事件：

- $1\ v\ x$  ( $1 \leq x \leq 10^5$ ) 表示在编号为 $v$ 的节点新出现了 $x$ 个蘑菇
- $2\ v$  表示克拉莉丝的起点变成了节点 $v$

在每个事件之后，克拉莉丝想要知道，他收集完所有的蘑菇所需的代价。

蘑菇的收集规则是这样的，对于每个蘑菇，克拉莉丝要收集它，所需要的代价是这个蘑菇所在节点和起点之间的路径上最靠近起点的边的边权。在起点上的蘑菇不需要代价就能收集。

### Input

第一行一个整数 $N$  ( $1 \leq N \leq 10^6$ )，表示节点数。

接下来 $N - 1$ 行，每行三个整数 $u, v, w$  ( $1 \leq w \leq 10^5$ )，表示在节点 $u$ 和 $v$ 之间有一条边权为 $w$ 的无向边。

接下来一行一个整数 $Q$  ( $1 \leq Q \leq 10^6$ )，表示事件的个数。

接下来 $Q$ 行，表示事件，格式如题面所示。

### Output

对于每个事件，输出一个整数，表示克拉莉丝至少需要多少时间才能采集完所有的蘑菇。

## Example

standard input	standard output
5	2
1 2 3	11
1 3 1	17
2 4 2	22
2 5 4	22
9	20
1 3 2	26
1 5 3	48
1 4 2	13
2 2	
1 2 5	
2 4	
1 1 3	
2 5	
2 3	

## Problem G. 糖糖王国的道路修建

Input file:           standard input  
Output file:          standard output  
Time limit:           1 second  
Memory limit:        256 megabytes

质量和辉辉是一对姐妹，他们统治着糖糖王国。糖糖王国有 $n + m$ 个城镇，可以看作是平面上的 $n + m$ 个整点，其中 $n$ 个属于质量，另外 $m$ 个属于辉辉。

现在质量和辉辉要在各自的城镇之间修建道路，道路必须是笔直的线段。学过简单图论的小盆友都知道，只需要 $n + m - 2$ 条线段就可以让质量和辉辉的城镇各自两两相连。

现在就希望你能给出方案，要求在满足连通性的同时，任意两条线段都不能在城镇点以外的地方相交。如果不存在这样的方案也请判断。

### Input

第一行读入两个整数 $n, m$ ，保证 $n > 1, m > 1$ 且 $n + m \leq 3000$ 。

接下来 $n$ 行读入 $n$ 个二维平面上的整点坐标 $x, y$ ，代表质量的 $n$ 个城市，编号为1到 $n$ 。

接下来 $m$ 行读入 $m$ 个二维平面上的整点坐标 $x, y$ ，代表辉辉的 $m$ 个城市，编号为1到 $m$ 。

保证在这 $n + m$ 个点中不存在重叠和三点共线，且横纵坐标为不超过 $10^9$ 的非负整数。

### Output

如果无解，请输出'Poor Quailty'，否则输出 $n + m - 2$ 行，前 $n - 1$ 行代表质量城镇的连接方式，之后 $m - 1$ 行代表辉辉城镇的连接方式。对于连接方式的每行请输出2个整数 $x, y$ 代表把 $x$ 和 $y$ 号城镇相连。

### Example

standard input	standard output
2 3	2 1
0 0	1 3
1 1	3 2
1 0	
0 1	
2 3	



## Problem H. 叁佰爱抠的序列

Input file: standard input

Output file: standard output

Time limit: 1 second

Memory limit: 256 megabytes

叁佰爱抠的智商有三百，他特别喜欢构造各种各样的序列。

这天他突然想出了这样一个序列：

一个长度为 $n$ 的序列 $A$ ，元素的值域是 $[1, m]$ 。

对于任意 $x, y \in [1, m], x \neq y$ ，序列中存在一个位置 $p(1 \leq p < n)$ ，满足 $A[p] = x, A[p+1] = y$ 或者 $A[p+1] = x, A[p] = y$ 。

叁佰爱抠很开心地走了。作为他忠实粉丝的你只听清楚了 $n$ 的大小，那么你好奇了起来， $m$ 最大值可以取到多少，并且打算自己动手构造一个这样的序列。

### Input

读入一个整数 $n(1 \leq n \leq 10^{18})$ ，代表序列的长度。

### Output

若 $n > 2 * 10^6$ ，请直接输出一个整数 $m$ 代表最大取值。

若 $n \leq 2 * 10^6$ ，请按以下格式输出：第一行输出一个整数 $m$ ，代表最大取值。第二行输出 $n$ 个整数，代表满足条件的序列。

### Examples

standard input	standard output
8	4 1 2 3 1 4 3 4 2
6	3 1 2 3 1 1 1

## Problem I. 堡堡的宝藏

Input file: standard input

Output file: standard output

Time limit: 3 seconds

Memory limit: 256 megabytes

堡堡有一张大小为 $n \times m$ 的地图，每个位置都有一个宝箱，宝箱要通过一定数量的投币才能打开。

堡堡想要打开所有的宝箱，但是他不想浪费太多钱。幸运的是，位置相邻的宝箱内部是相连的。

具体地说，存在 $k$ 个约束条件，每个约束条件为：（保证 $(x_1, y_1)$ 与 $(x_2, y_2)$ 相邻， $\text{coin}[x][y]$ 代表对宝箱 $(x, y)$ 投币数量）

$$\text{coin}[x_1][y_1] + \text{coin}[x_2][y_2] \geq w$$

。

请你求出堡堡最少需要投多少个币才能打开所有的宝箱。

### Input

第一行读入三个整数 $n, m, k$  ( $1 \leq n \times m \leq 1500, 0 \leq k \leq n \times m$ )。

接下来 $k$ 行读入 $x_1, y_1, x_2, y_2, w$  ( $1 \leq x_1, x_2 \leq n, 1 \leq y_1, y_2 \leq m, 0 \leq w \leq 10^9$ )。

### Output

输出一行一个整数代表答案。

### Example

standard input	standard output
2 2 2 1 1 1 2 1 1 1 2 1 1	1

## Problem J. 邦邦的2-SAT模板

Input file: standard input  
 Output file: standard output  
 Time limit: 1 second  
 Memory limit: 256 megabytes

邦邦是图论白痴，他有一天捡到了一份模板，可以解决2-SAT问题并输出方案。

所谓2-SAT问题，指：有 $n$ 个布尔变量 $a_i$ ，有 $m$ 个形如 $x \text{ or } y = \text{true}$ 的方程， $x$ 和 $y$ 为 $a_i$ 或者 $\neg a_i$ ，求是否存在一组 $a_i$ 的取值满足所有方程。

戳戳是真正的图论大师，他看了看邦邦的板子，发现这段代码会超时。邦邦不相信，戳戳要赶去约会了，于是希望你构造一个数据让邦邦这段代码超时。

具体地，你需要根据给定的 $n$ 按如下格式构造：

第一行输出一个整数 $m(0 \leq m \leq n)$ ，代表有 $m$ 个方程。

接下来 $m$ 行，给出两个数 $x$ 和 $y(-n \leq x, y \leq n, x, y \neq 0)$ ，若数字为负数，代表 $\neg a_i$ ，否则代表 $a_i$ 。

要求保证代码中solve的返回值是true(存在至少一组解)，且 $CNT$ 的值满足

$$n^2 \leq 2 * CNT$$

。

邦邦的2-SAT模板见附录。

### Input

读入一个整数 $n(1 \leq n \leq 3000)$ 。

### Output

按规定格式输出。

### Example

standard input	standard output
1	0

### Note

以下是邦邦的2-SAT模板：

```
#include<cstdio>
using namespace std;
const int N=3010;
int g[N<<1],nxt[N<<1],v[N<<1],num;
int q[N<<1],t;
bool vis[N<<1];
int CNT;
```

```
int n,m;
void add(int x,int y){
    nxt[++num]=g[x];
    v[num]=y;
    g[x]=num;
}
bool dfs(int x){
    CNT++;
    if(vis[x>n?x-n:x+n])return 0;
    if(vis[x])return 1;
    vis[q[++t]=x]=1;
    for(int i=g[x];i;i=nxt[i])if(!dfs(v[i]))return 0;
    return 1;
}
bool solve(){
    for(int i=1;i<=n;i++)if(!vis[i]&&!vis[i+n]){
        t=0;
        if(!dfs(i)){
            while(t)vis[q[t--]]=0;
            if(!dfs(i+n))return 0;
        }
    }
    return 1;
}
int main()
{
    scanf("%d%d",&n,&m);
    for(int i=1;i<=m;i++){
        int x,y;scanf("%d%d",&x,&y);
        if(x<0)x=n-x;if(y<0)y=n-y;
        add(x>n?x-n:x+n,y);add(y>n?y-n:y+n,x);
    }
    solve();
    return 0;
}
```

## Problem K. 破忒头的匿名信

Input file:           standard input  
Output file:         standard output  
Time limit:          2 seconds  
Memory limit:       256 megabytes

破忒头想要写一封匿名信来做坏事，由于他不想被认出自己的笔迹，因此他想要雇佣萨博来帮他写这封信。萨博按照这样的标准来收费：他的词典里有 $N$ 个单词 $S_1, S_2, \dots, S_N$ ，第 $i$ 个单词的单价是 $p_i$ 。如果你提供一个长度为 $M$ 的序列 $a_1, a_2, \dots, a_M (1 \leq a_i \leq N)$ ，那么你需要支付 $\sum_{i=1}^M p_{a_i}$ 的金钱，而萨博会依次往信里写上 $S_{a_1}, S_{a_2}, \dots, S_{a_M}$ 。破忒头希望支付最少的金钱，让萨博写的内容恰好为他想要的信件内容 $T$ 。请你告诉破忒头，最少需要付多少钱，能让萨博写出他想要的匿名信，或者告诉他这是不可能做到的。

### Input

对于每组测试数据，第一行包含一个正整数 $N$ ，表示萨博词典里的单词个数( $1 \leq N \leq 5 * 10^5$ )。接下来的 $N$ 行描述词典里的每个单词，第 $i$ 行包含一个字符串 $S_i$ 和这个字符串的单价 $p_i (1 \leq p_i \leq 10^9)$ 。接下来一行包含破忒头想要的信件内容 $T (1 \leq |T| \leq 5 * 10^5)$ 。保证 $S_i$ 和 $T$ 都仅包含小写字母，且 $S_i$ 的总长度不超过 $5 * 10^5$ 。

### Output

如果有可能让萨博写出破忒头想要的信件内容，那么输出一个正整数，表示最小需要付出的代价。否则，输出-1。

### Examples

standard input	standard output
4 ab 5 cd 10 abc 100 d 1 abcd	15
1 ab 1 abc	-1