Problem A: Python File

Time limit: **1 second**
Memory limit: **256 Megabytes**

A file name consists of 2 parts: base name and extension name, separated by a dot character. For example, Python source code files have extension name "py", documents have extension name "doc" or "docx". In Windows OS, file names are case insensitive.

Write a program to input a file name as string and check whether it is considered Python source code file in Windows or not.

Input

Only one line contain file name S ($1 \le |S| \le 128$). File name will only contain ‘a’-‘z’, ‘A’-‘Z’, ‘.’ and ‘_’.

Output

If file is a python file, print "yes", otherwise "no".

Sample Input 1	Sample Output 1
`abc.py`	`yes`

Sample Input 2	Sample Output 2
`abc.bin`	`no`

Problem B: Shoes Game

Time limit: **1 second**
Memory limit: **256 Megabytes**

N + 1 (*N* is odd number) shoes from *(N+1)/2* pair of shoes the same type with different sizes are lined up in arandom order. The game master secretly take one of the shoes out and hide it. The player need to guess whether the hidden shoe is for left foot or right foot and what is its size. Write a program to solve this game with cheating computer power.

Input

The first line of input contains the one integer N ($1 \le N \le 10^5$), the number of shoes. The following line contains *N* integers $S_1, S_2, ..., S_N$ $\left(1 \le |S_i| \le 10^9\right)$. Shoe *i* is for left foot if $S_i < 0$, otherwise it is for right foot. The size of the shoe is $|S_i|$.

Output

Output one integer *R* where $|R|$ is equal to the hidden shoe's size and it is negative if it is a shoe for left foot, positive otherwise.

Sample Input 1	Sample Output 1
3 1 5 -1	-5

Problem C: Coin Toss

Time limit: **1 second**
Memory limit: **256 Megabytes**

A sequence of coin toss can be encoded as a binary string, ‘0’ for head, ‘1’ for tail. Given an encoded sequence *S*, toss the coin until *S* appears in the result sequence then stop. Let *T* be the number of coin toss taken for *S* to appear, find the expected value of *T*.

Input

The first line of input contains the one integer Q ($1 \le Q \le 10^4$), the number of test cases. The following Q lines, each contain a string S ($1 \le |S| \le 20$).

Output

Output Q lines, each contains the respective value of *T*. The answer is considered correct if the precision error is less than 10^{-9}.

Sample Input 1	**Sample Output 1**
`1` `00`	`6`

Problem D: Binary String Set

Time limit: **1 second**
Memory limit: **256 Megabytes**

Let S be a set of strings. S^* is a set of the empty string and any concatenation of strings in S (each string in S can appear multiple times).

Given n strings $T_1, \ldots, T_n$. For each T_i, find the least number of characters to remove to satisfy $T_i \in \{0, 01, 10\}^*$.

Input

The first line of input contains one integer n ($1 \le n \le 10^6$). The following n lines, each contains a string T_i. The total length of all T_i will not exceed 10^6.

Output

Output n lines, one integer on each line show the number of characters to remove for the respective input.

Sample Input 1	Sample Output 1
2 00110 110	0 1

Problem E: Special Number

Time limit: **1 second**
Memory limit: **256 Megabytes**

A number is special if the sum of its digits is a prime number. Given an integer N, count the number of positive integer pairs (x, y) where both x and y are special and $x + 2y = N$.

Input

The first line of input contains the one integer N ($1 \le N \le 10^{15}$).

Output

Output a single number, the number of satisfied pairs.

Sample Input 1	Sample Output 1
`100`	`7`

Problem F: Greetting Card

Time limit: **1 second**
Memory limit: **256 Megabytes**

In a certain village, N households form circle. Initially, household i's happiness level is A_i ($0 \le i < N$). In the morning, each household will write their greetting cards to others but order them to only arrive at the recipients after work hour. After work, everyone is burnt out and their happiness level depends solely on the greetting card they receive. Therefore, their happiness level at the end of the day will be the sum of happiness level of the senders whom they receive from at the time the cards were written.

On day 0, each household write for their adjacent households only. On each following days, they double the distance which they send the card to. More specifically, on day t, household i will write to household $(i+2^t)\,\%\,N$ and $(i-2^t\,\%\,N+N)\,\%\,N$. One household may write for the same household twice on the same day.

Given an integer K, calculate everyone's happiness level after 2^K days (the end of day 2^K-1).

Input

The first line of input contains 2 integers N and K ($2 \le N \le 10^5$, $0 \le K \le 10^9$), the number of test cases.
The following line contains N integers A_i ($1 \le A_i \le 10^9$).

Output

Output one line contain N integers, represent the happiness level of the households (modulo 10^9+7) after K days respectively.

Sample Input 1	**Sample Output 1**
`3 0` `1 2 3`	`5 4 3`

Sample Input 2	**Sample Output 2**
`3 1` `1 2 3`	`7 8 9`

Problem G: Rounded Convex Hull

Time limit: **8 seconds**
Memory limit: **512 Megabytes**

On a 2D plane, there are N circles and M polygons. Find the perimeter of the convex hull of all these figures.

Input

The first line of input contains 2 integers N and M ($0 \le N, M \le 10^5$, $1 \le N + M$), the number of circles and polygons.

The next N lines, each has 3 numbers x_i, y_i and r_i which is the center of the circle and its radius $\left(|x_i|, |y_i| \le 5\times10^4,\ 0 \le r_i \le 5\times10^4\right)$.

The next M lines, each line start with an integer p_i ($p_i \ge 1$), the number of vertices in the polygon followed by p_i pair of numbers $(x_{i,1}, y_{i,1}), \ldots, (x_{i,p_i}, y_{i,p_i})$ $\left(|x_{i,j}|, |y_{i,j}| \le 5\times10^4\right)$.Total number of vertices on all polygons will not exceeed 10^5.

Output

Output the perimeter of the convex hull. The answer is considered correct if precision error is less than 10^{-5}.

Sample Input 1	**Sample Output 1**

`3 2`
`-14.123000 -1.456000 5.789000`
`0.123000 14.456000 4.789000`
`-6.868686 20.456780 3.789285`
`1 5.123000 5.456000`
`2 6.879000 6.123000 7.456000 7.789000`
	`88.888888`

Problem H: Grouping

Time limit: 1 second
Memory limit: **256 Megabytes**

There are $2n$ students in a school. Calculate the number of ways pick a group of at least 2 students so that for pair of students a and b in this group, $|a-b| \neq 1$ and $|a-b| \neq n+x$.

Input

The first line of input contains 2 integers n and x $\left(1 \leq n \leq 10^{18},\ 0 \leq x \leq n\right)$.

Output

Output a single integer, the answer to the problem modulo 10^9+7.

Explaination

In the sample test case, we can pick $\{1,3\}$ or $\{2,4\}$.

Sample Input 1	Sample Output 1
2 1	2

Problem I: Gift Box Lottery

Time limit: **3 seconds**
Memory limit: **512 Megabytes**

Bob takes part in a traditional gift box lottery game held every year. In this game, there are an infinite number of identical gift boxes with the same content inside. Each box contains N gifts, labeled from 1 to N, have value $A_1, A_2, ..., A_N$ respectively, the players know these values ahead of t ime. A player must start out with a random gift box, pulling gifts from it randomly, one gift at a time. All gifts in the box have an equal chance to be pulled. The gift with label 1 is special, after you pull this gift, you gain the right to skip remaining gifts in this box and move on to the next gift box. This right can be exercised whenever, as long as you are still pulling from the same box. Unless using the above special right, you must pull until the current box is empty before moving to the next box. Pulling special gift from the next box will grant you special right for the new box.

If Bob plays this game optimally, what would be the best value per pull he can get?

Input

The first line of input contains one integer N ($1 \le N < 20$), the number of gifts per gift box. The following line contains N real numbers A_i ($0 \le A_i \le 10^6$), the value of each gifts.

Output

Output a single number, the best value per pull Bob can achieve if he plays optimally. The answer is considered correct if the precision error is less than 10^{-5}.

Explaination

In the first sample test case, no matter which gift we pull first from the box, we still need to pick the other one before moving to the next box, so the result is $1/2 = 0.5$.

In the second sample test case, in case we:

• Pull gift 1 first, move on to next box immediately, result in 1 value from 1 pull.

• Pull gift 2 first, we pull again then move on to next box, result in 1 value from 2 pulls. Combined, the result is $2/3$.

Sample Input 1	Sample Output 1
2 0 1	0.5

Sample Input 2	Sample Output 2
2 1 0	0.6666666667

Sample Input 3	Sample Output 3
3 2 1 0	1.25

Problem J: Workers Roadmap

Time limit: **1 second**
Memory limit: **256 Megabytes**

A business received a big order for M products, they must deliver as soon as possible but there is nothing left in the warehouse, they have to produce from scratch. They have N workers, worker i produce A_i products every day but take 1 day of leave after every B_i days of work. Calculate what's the earliest day they can finish producing M products to deliver.

Input

The first line of input contains 2 integers N and M $\left(1 \le N \le 100,\ 1 \le M \le 10^{15}\right)$, the number of workers and number of products ordered.

The next N lines, each has 2 numbers A_i and B_i $\left(1 \le A_i, B_i \le 10^{15}\right)$.

Output

Output one integer, the number of days it takes.

Sample Input 1	**Sample Output 1**

`2 30`
`2 5`
`1 9`
	`11`

Problem K: Lexigraphical Matrix

Time limit: **1 second**
Memory limit: **256 Megabytes**

A Lex Matrix is a matrix of size $m \times n$, m rows, n columns. Rows are numbered from 1 to m, top to bottom. Columns are numbered from 1 to n, left to right. $A_{x,y}$ is the y-th value on row x. Each row is a permutation of $1, 2, ..., n$.

Lex Matrix A is considered greater than Lex Matrix B if compare each cell starting from the first row, left to right then to the next row and so on, the first pair of cells (i, j) where $A_{i,j} \neq B_{i,j}$, $A_{i,j} > B_{i,j}$ hold.

Given a Lex Matrix A. You are allowed to pick 2 rows/columns, swap them and repeat by picking again as many times as you want, modify to achieve the greatest possible Lex Matrix from A. Let's call this maximal matrix A'. Given q pairs of number x_i and y_i, $(1 \le x_i \le m,\ 1 \le y_i \le n)$, find the value of A'_{x_i,y_i}.

Input

The first line of input contains 2 integers m and n ($1 \le m, n \le 500$).

The next m lines, representing Lex Matrix A, each contains n numbers, a permutation of $1, 2, ..., n$.

The next line contains one integer q ($1 \le q \le 10000$).

The next q lines, each contains 2 integers x_i and y_i $(1 \le x_i \le m,\ 1 \le y_i \le n)$.

Output

Output q lines, each contains one integer, the value of A'_{x_i,y_i}.

Sample Input 1	Sample Output 1
2 3 1 2 3 1 2 3 2 1 1 2 3	3 1

Problem L: Subset Counting

Time limit: **1 second**
Memory limit: **256 Megabytes**

Given a sequence of integers $a_1, a_2, ..., a_n$; find the number of sets S satisfying the following conditions:

- $S \subset \{1, 2, ..., n\}$
- $\exists x \in S : a_x \notin S$
- $\exists y \in S : (\forall x \in S : a_x \neq y)$

Since the result can be rather large, you should output it modulo 998244353.

Input
The input contains multiple test cases. Each test case is presented in two lines as below:

- The first line contains an integer n ($1 \leq n \leq 10^5$).
- The second line contains n integers $a_1, a_2, ..., a_n$ ($1 \leq |a_i| \leq n$).

The input is terminated by a line with a single integer 0 which is not a test case. The sum of n over all test cases does not exceed 10^6.

Output
For each test case, write the result on the single line.

Explaination
In the second test cases, 6 valid sets are $\{1\}$; $\{2\}$; $\{3\}$; $\{1, 2\}$; $\{2, 3\}$; $\{3, 1\}$.

Sample Input 1	**Sample Output 1**
3 1 2 3 3 2 3 1 0	0 6

Problem M: Graph

Time limit: **1 second**

Memory limit: **256 Megabytes**

You are given a graph containing n vertices and m directed arcs. Vertices are numbered from 1 to n, inclusive; and arcs are numbered from 1 to m, inclusive. The i-th arc starts from the s_i-th vertex, ends at the t_i-th vertex and has a cost of c_i.

In this problem, a **valid path** is a sequence of arcs in which the first one starts at the 1-st vertex, the last one ends at the n-th vertex and every arc ends at the starting vertex of the next one. More formally, a valid path can be represented by a sequence of indices ($e_1, e_2, ..., e_k$) satisfying all below conditions:

- $1 \le e_1, e_2, ..., e_k \le m$
- $s_{e_1} = 1$
- $s_{e_k} = n$
- $t_{e_j} = s_{e_{j+1}}$ for all j such that $0 < j < k$

Please note that these indices do not need to be distinct, meaning that a valid path may walk through some edge multiple times.

Your task is to mark some (possibly none or all) arcs of the given graph special so as to fullfill this requirements: Every valid path of this graph passes through special arcs **exactly once**. In other words, if the sequence of indices ($f_1, f_2, ..., f_k$) represents a valid path, there should be exactly one index l such that $1 \le l \le k$ and the f_l-th arc is marked special. Moreover, the total cost of special arcs should be as small as possible.

Input

The input contains multiple test cases. Each test case is presented as below:

- The first line contains two integers n and m $(2 \le n \le 100,\ 1 \le m \le 2500)$ denoting the number of vertices and arcs of the graph, respectively.
- In the next m lines, each contains three integers s_i, t_i and c_i $\left(1 \le s_i, t_i \le n,\ 1 \le c_i \le 10^9\right)$ denoting the starting and ending vertices of the i-th arc and its cost, respectively.

- The last line is a blank line.

The input is terminated by two zeros which do not represent a test case. It is guaranteed that:

- For every arc, its starting vertex differs from its ending one.
- For every graph, each ordered pair (s_i, t_i) appears at most once.
- A valid path always exists.

The sum of n over all test cases does not exceed 1000. The sum of m over all test cases does not exceed 25000.

Output

Write the result of each test case in a single line: If it is impossible to mark arcs in order to satisfy the above require ments, print "IMPOSSIBLE" (without quotes). Otherwise, print the minimum possible total cost of special arcs.

Explaination

In the first test case, there are three valid paths, whose arc indices are $(4,3)$, $(1,5)$ and $(1,2,3)$. The optimal solution is to mark the first and the fourth arcs special. Note that marking the first and the third arcs is not a valid way, since the valid path $(1,2,3)$ will pass through special arcs twice.

In the second test case, please be aware that a valid path can contain duplicated arcs. Hence, $(2,1,2)$ or even $(2,1,2,1,2)$ are valid paths of this graph. Consequently, marking the second arc is not possible, as it appears more than once in those valid paths. As a result, a solution does not exist.

Sample Input 1	**Sample Output 1**
4 5 1 2 1 2 3 1 3 4 1 1 3 8 2 4 8 2 2 2 1 1 1 2 1 0 0	9 IMPOSSIBLE