Problem A. A Post Robot

Description

DT is a big fan of digital products. He writes posts about technological products almost everyday in his blog.

But there is such few comments of his posts that he feels depressed all the day. As his best friend and an excellent programmer, DT asked you to help make his blog look more popular. He is so warm that you have no idea how to refuse. But you are unwilling to read all of his boring posts word by word. So you decided to write a script to comment below his posts automatically.

After observation, you found words "Apple" appear everywhere in his posts. After your counting, you concluded that "Apple", "iPhone", "iPod", "iPad" are the most high-frequency words in his blog. Once one of these words were read by your smart script, it will make a comment "MAI MAI MAI!", and go on reading the post.

In order to make it more funny, you, as a fan of Sony, also want to make some comments about Sony. So you want to add a new rule to the script: make a comment "SONY DAFA IS GOOD!" when "Sony" appears.

Input

A blog article described above, which contains only printable characters(whose ASCII code is between 32 and 127), CR(ASCII code 13, '`\r`' in C/C++), LF(ASCII code 10, '`\n`' in C/C++), please process input until EOF. Note all characters are **case sensitive**.

The size of the article does not exceed 8KB.

Output

Output should contains comments generated by your script, one per line.

Samples

Sample Input	Sample Output

`Apple bananaiPad lemon ApplepiSony`
`233`
`Tim cook is doubi from Apple`
`iPhoneipad`
`iPhone30 is so biiiiiiig Microsoft`
`makes good App.`
	`MAI MAI MAI!`
	`MAI MAI MAI!`
	`MAI MAI MAI!`
	`SONY DAFA IS GOOD!`
	`MAI MAI MAI!`
	`MAI MAI MAI!`
	`MAI MAI MAI!`

Problem B. Boring String Problem

Description

In this problem, you are given a string s and q queries.

For each query, you should answer that when all distinct substrings of string s were sorted lexicographically, which one is the k-th smallest.

A substring $s_{i...j}$ of the string $s = a_1a_2 \ldots a_n (1 \leq i \leq j \leq n)$ is the string $a_i a_{i+1} \ldots a_j$. Two substrings $s_{x...y}$ and $s_{z...w}$ are cosidered to be distinct if $s_{x...y} \neq s_{z...w}$

Input

The input consists of multiple test cases.Please process till EOF.

Each test case begins with a line containing a string $s(|s| \leq 10^5)$ with only lowercase letters.

Next line contains a postive integer $q(1 \leq q \leq 10^5)$, the number of questions.

q queries are given in the next q lines. Every line contains an integer v. You should calculate the k by $k = (l \oplus r \oplus v) + 1(l$, r is the output of previous question, at the beginning of each case $l = r = 0$, $0 < k < 2^{63}$, "$\oplus$" denotes exclusive-or)

Output

For each test case, output consists of q lines, the i-th line contains two integers l, r which is the answer to the i-th query. (The answer l, r satisfies that $s_{l...r}$ is the k-th smallest and if there are several l, r available, ouput l, r which with the smallest l. If there is no l, r satisfied, output "0 0". Note that $s_{1...n}$ is the whole string)

Samples

Sample Input	Sample Output
aaa 4 0 2 3 5	1 1 1 3 1 2 0 0

Problem C. Paint Pearls

Description

Lee has a string of n pearls. In the beginning, all the pearls have no color. He plans to color the pearls to make it more fascinating. He drew his ideal pattern of the string on a paper and asks for your help.

In each operation, he selects some continuous pearls and all these pearls will be painted to **their target colors**. When he paints a string which has k different target colors, Lee will cost k^2 points.

Now, Lee wants to cost as few as possible to get his ideal string. You should tell him the minimal cost.

Input

There are multiple test cases. Please process till EOF.

For each test case, the first line contains an integer $n (1 \leq n \leq 5 \times 10^4)$, indicating the number of pearls. The second line contains $a_1, a_2, \ldots, a_n$ $(1 \leq a_i \leq 10^9)$ indicating the target color of each pearl.

Output

For each test case, output the minimal cost in a line.

Samples

Sample Input	Sample Output
3 1 3 3 10 3 4 2 4 4 2 4 3 2 2	2 7

Problem D. Get the Nut

Description

Squirrely has lost his lifetime collection of acorns in an unfortunate geyser accident and now he needs your help! Roll the animals around and get Squirrely to the Acorn, but beware! Not all animals are fluffy as they appear to be. Start from your home forest, and continue through the harsh weather of the snow, the scary swamp and the vast wild west! In your journey you will encounter great dangers, but don't worry! You and Squirrely will be a great team.

Get the Nut is a very cute game as mentioned above. Now Squirrely is in a forest, which can be divided into 6 rows and 8 columns, 48 grids in total. Squirrely wants to get the nut in another grid. However, it is much more difficult than you can image because the forest is full of danger. There are two kinds of other animal in the forest: Mouse and Pig. Pigs are very vicious, they will kill and eat any other animal (except Pigs, of course) in the adjacent grids. Here "adjacent" means two grids share a common edge. Once any other animal enter grids adjacent to a Pig, it will stop and be eaten. Once an animal is adjacent to the nut, it will stop and eat the nut. So if Squirrely is eaten by a Pig or the nut is eaten by a Pig or a Mouse, the game is over and you fail. At the beginning of the game, each animal occupies one grid. For each step, you can choose one animal to roll in either up, down, left or right direction, the animal will keep rolling until it reach the border of the forest or hit a tree or an animal. Your task is to calculate the minimal number of steps should be made that Squirrely can get the nut.

Input

There are multiple test cases. Please process till EOF.

Each test case gives the map of the forest, which has 6 rows and 8 columns. Each grid of the map is one of the following six characters:

1. '`.`' - the empty area which every animal can stay or go through
2. '`#`' - the tree which is an obstacle
3. '`S`' - Squirrely
4. '`M`' - a Mouse
5. '`P`' - a Pig
6. '`N`' - the nut which Squirrely want to get

You should notice that there is one blank line before every test case. You may assume that there is one and only one nut in the forest and the number of animals in the forest will not exceed 5, and the input guarantees that there are at most 32 empty grids(including the original grids occupied by animals) in the map.

Output

For each test case, output a positive integer indicating the desired answer. You may assume that there is always exists a solution to get the nut, and the minimal number of steps should be made will not exceed 30.

Samples

Sample Input	Sample Output
#......N ###.#### ###M#### ###.#### ###.#### ###S#### ######## ###.#### ##.....# #S.M.P.N ######## ######## ######## S...M.## ####M.## ####M.## ###P..## ####N.## P.....M# ...##..# P..##..# .###...# ..N#...# .###..S#	4 4 5 15

Sample Explanations

Case #1:

1. roll the Mouse at grid (3, 4) up to grid (1, 4).
2. roll the Mouse at grid (1, 4) left to grid (1, 2).
3. roll Squirrely at grid (6, 4) up to grid (1, 4).
4. roll Squirrely at grid (1, 4) right to grid(1, 7) and he can get the nut.

Case #2:

1. roll the Pig at grid (4, 6) up to grid (3, 6).
2. roll the Pig at grid (3, 6) left to grid (3, 4), why grid (3, 4)? Because it will stop to eat the Mouse at grid (4, 4).

3. roll the Pig at grid (3, 4) up to grid (2, 4), you must roll the Pig to grid (2, 4), otherwise, Squirrely will certainly be eaten by the Pig!
4. just roll Squirrely at grid (4, 2) right to grid (4, 7) and he can get the nut.

Case #3:

1. roll the Mouse at grid (4, 5) down to grid (5, 5), and it is eaten by the Pig, notice that the Mouse at grid(5, 5) can not eat the nut because the Mouse will be eaten by the Pig before it can eat the nut!
2. roll the Mouse at grid (3, 5) down.
3. roll the Mouse at grid (2, 5) down.
4. roll Squirrely at grid (2, 1) right to grid (2, 6).
5. roll Squirrely at grid (2, 6) down to grid (6, 6) and he can get the nut.

Hint

1. When a Mouse or Squirrely is adjacent to a Pig and a nut at the same time, it will be eaten by the Pig first.
2. At the beginning of the game, it's guaranteed that you can't win or fail at once.

Problem E. Game

Description

Here is a game for two players. The rule of the game is described below:

- In the beginning of the game, there are a lot of piles of beads.
- Players take turns to play. Each turn, player choose a pile i and remove some (at least one) beads from it. Then he could do nothing or split pile i into two piles with a beads and b beads.($a, b > 0$ and $a + b$ equals to the number of beads of pile i after removing)
- If after a player's turn, there is no beads left, the player is the winner.

Suppose that the two players are all very clever and they will use optimal game strategies. Your job is to tell whether the player who plays first can win the game.

Input

There are multiple test cases. Please process till EOF.

For each test case, the first line contains a postive integer $n(n \leq 10^5)$ means there are n piles of beads. The next line contains n postive integers, the i-th postive integer $a_i(a_i < 2^{31})$ means there are a_i beads in the i-th pile.

Output

For each test case, if the first player can win the game, ouput "Win" and if he can't, ouput "Lose".

Samples

Sample Input	Sample Output
1 1 2 1 1 3 1 2 3	Win Lose Lose

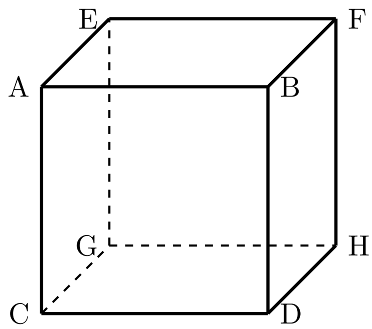
Problem F. Dice

Description

There are 2 special dices on the table. On each face of the dice, a distinct number was written. Consider $a_1.a_2, a_3, a_4, a_5, a_6$ to be numbers written on top face, bottom face, left face, right face, front face and back face of dice A. Similarly, consider $b_1.b_2, b_3, b_4, b_5, b_6$ to be numbers on specific faces of dice B. It's guaranteed that all numbers written on dices are integers no smaller than 1 and no more than 6 while $a_i \neq a_j$ and $b_i \neq b_j$ for all $i \neq j$. Specially, sum of numbers on opposite faces may not be 7.

At the beginning, the two dices may face different(which means there exist some i, $a_i \neq b_i$). Ddy wants to make the two dices look the same from all directions(which means for all i, $a_i = b_i$) only by the following four rotation operations.(Please read the picture for more information)

- Left Rotation: rotate the dice by 90 degrees on $\overline{CG}$ after which $\overline{ACGE}$ becomes the bottom face.
- Right Rotation: rotate the dice by 90 degrees on $\overline{DH}$ after which $\overline{BDHF}$ becomes the bottom face.
- Front Rotation: rotate the dice by 90 degrees on $\overline{CD}$ after which $\overline{ABCD}$ becomes the bottom face.
- Back Rotation: rotate the dice by 90 degrees on $\overline{GH}$ after which $\overline{EFGH}$ becomes the bottom face.



Now Ddy wants to calculate the minimal steps that he has to take to achieve his goal.

Input

There are multiple test cases. Please process till EOF.

For each case, the first line consists of six integers $a_1, a_2, a_3, a_4, a_5, a_6$, representing the numbers on dice A.

The second line consists of six integers $b_1, b_2, b_3, b_4, b_5, b_6$, representing the numbers on dice B.

Output

For each test case, print a line with a number representing the answer. If there's no way to make two dices exactly the same, output -1.

Samples

Sample Input	Sample Output
1 2 3 4 5 6 1 2 3 4 5 6 1 2 3 4 5 6 1 2 5 6 4 3 1 2 3 4 5 6 1 4 2 5 3 6	0 3 -1

Problem G. City Tour

Description

There are m visitors coming to visit country A, and they plan to visit all n cities in the country one after another. The cities are numbered from 1 to n by the order they are visited. The visitors start their tour at city 1. Each day, for each visitor i, he has p_i probability to go to next city(which means city number increases by 1), and $1 - p_i$ probability to fall in love with current city and stay there till the end of tour. If a visitor reach city n, he will not move any more.

When visitor i reach city j, he get h_{ij} units of happiness. For $j > 1$, suppose city j is visited by $c_j(c_j > 0)$ visitors and city $j-1$ is visited by $c_{j-1}(c_{j-1} > 0)$ visitors, then each of city j's visitors will get extra $\frac{c_j}{c_{j-1}}h_{ij}$ units of happiness.

Let h_{tot} denote the total happiness of all visitors at the end of tour. Now you need to calculate the expectation of h_{tot}.

Input

There are multiple test cases. Please process till EOF.

For each case, the first line contains two integers m and n $(1 \le m, n \le 16)$, indicating the number of visitors and the number of cities respectively.

The second line contains m real numbers $p_i(0 \le p_i \le 1)$—the probability for the i-th visitor to move to next city each day. The probabilities are given with at most 6 digits after decimal point.

Then there are m lines follow, each line contains n integers. The j-th integer of ith line denotes h_{ij} $(1 \le h_{ij} \le 100)$.

Output

For each test case, print a single real number in a line, represents the expectation of h_{tot}. The answer will be considered valid if it differs from the correct one by at most 10^{-5}.

Samples

Sample Input	Sample Output
3 1 0.1 0.2 0.3 10 20 30 3 3 0.5 0.5 0.5 1 1 1 1 1 1 1 1 1 4 4 0.1 0.4 0.2 0.3 7 2 18 10 2 6 9 5 4 4 19 17 7 3 13 17	60.000000 6.8437500 34.23064559

Problem H. Number Sequence

Description

There is a special number sequence which has $n+1$ integers. For each number in sequence, we have two rules:

- $a_i \in [0, n]$
- $a_i \neq a_j (i \neq j)$

For sequence a and sequence b, the integrating degree t is defined as follows("$\oplus$" denotes exclusive-or):

$$t = (a_0 \oplus b_0) + (a_1 \oplus b_1) + \cdots + (a_n \oplus b_n)$$

(sequence B should also satisfies the rules described above)

Now give you a number n and the sequence a. You should calculate the maximum integrating degree t and print the sequence b.

Input

There are multiple test cases. Please process till EOF.

For each case, the first line contains an integer $n (1 \leq n \leq 10^5)$, The second line contains $a_0, a_1, a_2, \ldots, a_n$.

Output

For each case, output two lines. The first line contains the maximum integrating degree t. The second line contains $n+1$ integers $b_0, b_1, b_2, \ldots, b_n$. There is exactly one space between b_i and $b_{i+1} (1 \leq i \leq n-1)$. Don't ouput any spaces after b_n.

Samples

Sample Input	Sample Output
4 2 0 1 4 3	20 1 0 2 3 4

Problem I. 233 Matrix

Description

In our daily life we often use 233 to express our feelings. Actually, we may say 2333, 23333, or 233333 ... in the same meaning. And here is the question: Suppose we have a matrix called 233 matrix. In the first line, it would be $233, 2333, 23333\dots$ (it means $a_{0,1} = 233, a_{0,2} = 2333, a_{0,3} = 23333\dots$) Besides, in 233 matrix, we got $a_{i,j} = a_{i-1,j} + a_{i,j-1} (i, j \neq 0)$. Now you have known $a_{1,0}, a_{2,0}, \dots, a_{n,0}$, could you tell me $a_{n,m}$ in the 233 matrix?

Input

There are multiple test cases. Please process till EOF.

For each case, the first line contains two postive integers $n, m (n \leq 10, m \leq 10^9)$. The second line contains n integers, $a_{1,0}, a_{2,0}, \dots, a_{n,0} (0 \leq a_{i,0} < 2^{31})$.

Output

For each case, output $a_{n,m}$ mod 10000007.

Samples

Sample Input	Sample Output
1 1 1 2 2 0 0 3 7 23 47 16	234 2799 72937

Sample Explanations

Case #1:

$$\begin{pmatrix} 0 & 233 \\ 1 & 234 \end{pmatrix}$$

Case #2:

$$\begin{pmatrix} 0 & 233 & 2333 \\ 0 & 233 & 2566 \\ 0 & 233 & 2799 \end{pmatrix}$$

Problem J. Mart Master II

Description

Trader Dogy lives in city S, which consists of n districts. There are $n-1$ bidirectional roads in city S, each connects a pair of districts. Indeed, city S is connected, i.e. people can travel between every pair of districts by roads.

In some districts there are marts founded by Dogy's competitors. when people go to marts, they'll choose the nearest one. In cases there are more than one nearest marts, they'll choose the one with minimal city number.

Dogy's money could support him to build only **one** new mart, he wants to attract as many people as possible, that is, to build his mart in some way that maximize the number of people who will choose his mart as favorite. Could you help him?

Input

There are multiple test cases. Please process till EOF.

In each test case:

First line: an integer n indicating the number of districts.

Next n-1 lines: each contains three numbers b_i, e_i and w_i, $(1 \leq b_i, e_i \leq n, 1 \leq w_i \leq 10000)$, indicates that there's one road connecting city b_i and e_i, and its length is w_i.

Last line : $n(1 \leq n \leq 10^5)$ numbers, each number is either 0 or 1, i-th number is 1 indicates that the i-th district has mart in the beginning and vice versa.

Output

For each test case, output one number, denotes the number of people you can attract, taking district as a unit.

Samples

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

Problem K. Ellipsoid

Description

Given a 3-dimension ellipsoid

$$ax^2 + by^2 + cz^2 + dyz + exz + fxy = 1$$

your task is to find the minimal distance between the original point $(0, 0, 0)$ and points on the ellipsoid. The distance between two points (x_1, y_1, z_1) and (x_2, y_2, z_2) is defined as $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$.

Input

There are multiple test cases. Please process till EOF.

For each testcase, one line contains 6 real number $a, b, c(0 < a, b, c, < 1), d, e, f(0 \le e, f, g < 1)$, as described above. **It is guaranteed that the input data forms a ellipsoid**. All numbers are fit in double.

Output

For each test contains one line. A real number describes the minimal distance. Answer will be considered as correct if their absolute error is less than 10^{-5}.

Samples

Sample Input	Sample Output
`1 0.04 0.01 0 0 0`	`1.0000000`