Problem A
All is Well


Photo by Kalle Gustafsson

"All is well!" cries the watchman as he makes his way through the city. Each night he patrols the city, following a route that ensures that the people of the town hear him. You feel safe and secure each night as you start to fall asleep and hear the familiar call from the street below.

The watchman is tired of walking, and has asked you to find a shorter route. He has a map of the city, laid out as a grid, where each cell represents an area of 1 by 1 kilometer. Some cells are where people live, and the watchman cannot enter those. The other cells take him different amounts of time to enter. He always starts his route by entering the upper-left cell (unless a person is in that cell, in which case he is unable to make any route).

The people are identified using uppercase English letters A–Z, and need to be visited in that order. The watchman knows where each person lives, and also how close he needs to be to them so they can hear him. He is excellent at yelling: as long as he is within hearing range of a person, he can yell and make sure that only that person hears him, even if he is within hearing range of other people.

A person can hear the watchman if the watchman enters a cell on the map such that the Euclidean distance from the center of the watchman's cell to the center of the person's cell is less than or equal to the person's hearing range. If the number of rows between the watchman and person is dr and the number of columns between the watchman and person is dc, then the distance between the watchman and the person is $\sqrt{dr^2 + dc^2}$.

Please help the watchman find an efficient route!

Input

The input starts with a line with three integers r, c, and p where $1 \leq r, c \leq 100$ and $1 \leq p \leq 26$. The values r and c are the number of rows and columns in the grid representing the city and p is the number of people who need to hear the watchman. They are identified by the first p uppercase letters of the English alphabet (the first person is A, the second is B, and so on).

The next p lines describe the hearing ranges of those people, in that same order. Each line has an integer $1 \leq h_i \leq 15$ indicating the range that person i can hear (in kilometers).

The next r lines describe the map. Each line has c space-separated tokens. Each token is either an integer $0 \leq w \leq 1\,000$ or a uppercase English letter. If the value is an integer, it is the amount of time (in minutes) it takes to enter that cell of the grid. If the value is a letter, it indicates where the person identified by that letter is located.

Output

If a route can be found, output the least amount of time (in minutes) that it takes for the watchman to have the people hear him in alphabetic order. If there is no valid route, output -1.

Sample Input 1

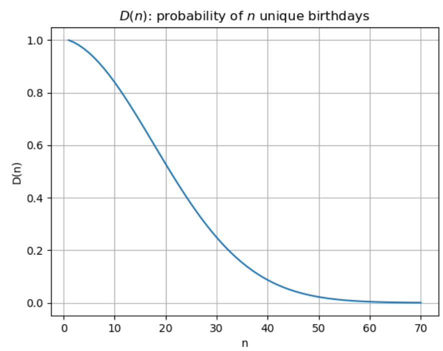
```
10 10 2
1
3
1 10 10 10 10 10 10 10 10 10
1 10 10 10 10 10 10 10 10 10
1 10 10 10 10 10 10 10 10 10
1 10 10 10 10 10 10 10 10 10
1 10 10 10 1 1 1 1 10 10
1 10 10 10 1 10 10 10 10 10
1 10 10 10 1 10 1 1 1 10
1 10 10 10 1 10 10 10 1 10
1 10 10 10 1 10 10 10 1 1
1 1 1 1 1 1 10 10 B A
```

Sample Output 1

```
33
```

Problem B
Birthday Paradox

The Birthday Paradox is the name given to the surprising fact that if there are just 23 people in a group, there is a greater than 50% chance that a pair of them share the same birthday. The underlying assumptions for this are that all birthdays are equally likely (which isn't quite true), the year has exactly 365 days (which also isn't true), and the people in the group are uniformly randomly selected (which is a somewhat strange premise). For this problem, we'll accept these assumptions.



The probability of n unique birthdays among n people.

Consider what we might observe if we randomly select groups of $P = 10$ people. Once we have chosen a group, we break them up into subgroups based on shared birthdays. Among *many* other possibilities, we might observe the following distributions of shared birthdays:

- all 10 have different birthdays, or
- all 10 have the same birthday, or
- 3 people have the same birthday, 2 other people have the same birthday (on a different day), and the remaining 5 all have different birthdays.

Of course, these distributions have different probabilities of occurring.

Your job is to calculate this probability for a given distribution of people sharing birthdays. That is, if there are P people in a group, how probable is the given distribution of shared birthdays (among all possible distributions for P people chosen uniformly at random)?

Input

The first line gives a number n where $1 \leq n \leq 365$. The second line contain integers c_1 through c_n, where $1 \leq c_i \leq 100$ for all c_i. The value c_i represents the number of people who share a certain birthday (and whose birthday is distinct from the birthdays of everyone else in the group).

Output

Compute the probability b of observing a group of people with the given distribution of shared birthdays. Since b may be quite small, output instead $\log_{10}(b)$. Your submission's answer is considered correct if it has an absolute or relative error of at most 10^{-6} from the judge's answer.

Explanations

The first sample case shows $P = 2$ people with distinct birthdays. The probability of this occurring is $b = 364/365 \approx 0.9972602740$, and $\log_{10}(b) \approx -0.001191480807419$.

The second sample case represents the third example in the list given earlier with $P = 10$ people. In this case, the probability is $b \approx 0.0000489086$, and $\log_{10}(b) \approx -4.310614508857128$.

Sample Input 1	Sample Output 1
2 1 1	-0.001191480807419

Sample Input 2	Sample Output 2
7 1 1 2 1 3 1 1	-4.310614508857128

Problem C
Code Names

You are given W, a set of N words that are anagrams of each other. There are no duplicate letters in any word. A set of words $S \subseteq W$ is called "swap-free" if there is no way to turn a word $x \in S$ into another word $y \in S$ by swapping only a single pair of (not necessarily adjacent) letters in x. Find the size of the largest swap-free set S chosen from the given set W.

Input

The first line of input contains an integer N ($1 \leq N \leq 500$). Following that are N lines each with a single word. Every word contains only lowercase English letters and no duplicate letters. All N words are unique, have at least one letter, and every word is an anagram of every other word.

Output

Output the size of the largest swap-free set.

Sample Input 1	Sample Output 1
6 abc acb cab cba bac bca	3

Sample Input 2	Sample Output 2
11 alerts alters artels estral laster ratels salter slater staler stelar talers	8

Sample Input 3	Sample Output 3
6 ates east eats etas sate teas	4

Problem D
Flipping Patties


Photo by Lets Ideas Complete

Business is booming at Big Bill's Burger Shack. Bill sells the most convenient, customizable burger in town. Using either the shack's website or mobile application, customers can request patties that are prepared precisely *to the second*. The i-th order consists of two integers: d_i, the amount of time each side of the patty should be cooked and t_i, the exact time at which the customer would like to pick up their order.

Big Bill is a talented grill cook, and he only hires talented grill cooks. In a given second, a grill cook can use a hand to do at most one of the following:

1. place a new patty on the grill,
2. flip a patty that is already on the grill,
3. remove a patty from the grill to serve it to a customer, or
4. do nothing.

Each grill cook has two hands, so they can do up to two of the above actions in a second.

When a patty is placed or flipped on the grill, the side on the grill immediately starts cooking. For example, if a patty is placed on the grill at 500 seconds, flipped at 600 seconds, and removed at 700 seconds, then it has cooked 100 seconds on each side.

These customers are demanding — they want to be served exactly at t_i, and the patty must be cooked for exactly d_i seconds on each side. Once a patty is on the grill, it must not be taken off until the customer wants it served. Orders that are not prepared exactly to the customers' specifications would ruin Big Bill's reputation. However, he doesn't want to hire more grill cooks than he needs to. What is the minimum number of grill cooks needed to serve all the orders?

Input

The first line contains a single integer $1 \leq n \leq 100$, the number of orders for Big Bill's patties. The next n lines each contain two integers, d_i and t_i. This means that the i-th patty should be cooked for exactly d_i seconds on each side and needs to be served exactly at time t_i. It is guaranteed that $1 \leq d_i \leq 600$, $2 \leq t_i \leq 43\,200$, and that it is possible to cook every order to specification without taking an action before second 0.

Output

Output a single integer, the minimum number of cooks needed to prepare all of the orders.

Sample Input 1	Sample Output 1
3 1 4 2 5 7 14	1

Sample Input 2	Sample Output 2
4 2 4 2 4 2 4 2 4	2

Problem E
Full Depth Morning Show

All boring tree-shaped lands are alike, while all exciting tree-shaped lands are exciting in their own special ways. What makes Treeland more exciting than the other tree-shaped lands are the raddest radio hosts in the local area: Root and Leaf. Every morning on FM 32.33 (repeating of course), Root and Leaf of The Full Depth Morning Show serve up the hottest celebrity gossip and traffic updates.

The region of Treeland is made of n cities, connected by $n-1$ roads such that between every pair of cities there is exactly one simple path. The ith road connects cities u_i and v_i, and has a toll of w_i.

To reward their loyal listeners, The Full Depth Morning Show is giving away a number of travel packages! Root and Leaf will choose $n-1$ lucky residents from the city that sends them the most fan mail. Each of those residents then gets a distinct ticket to a different city in Treeland.

Each city in Treeland has its own tax on prizes: t_i. Let $d_{u,v}$ be the sum of the tolls on each road on the only simple path from city u to v. For a trip from city u to city v, the cost of that trip is then $(t_u + t_v)d_{u,v}$.

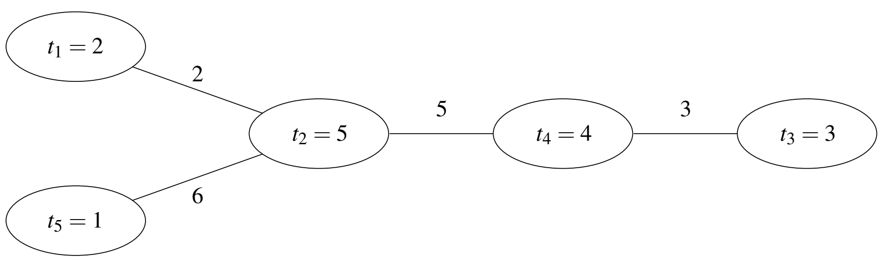


Figure E.1: The map of Treeland corresponding to the first sample input.

The shock jocks haven't quite thought through how much their prize is worth. They need to prepare a report to the radio executives, to summarize the expected costs. For each city that could win the prize, what is the total cost of purchasing all the tickets?

Input

The first line of input is a single integer n ($1 \le n \le 100\,000$). The next line has n space-separated integers t_i ($1 \le t_i \le 1\,000$), the tax in each city. The following $n-1$ lines each have 3 integers, u_i, v_i, w_i, meaning the ith road connects cities u_i and v_i ($1 \le u_i, v_i \le n$), with a toll of w_i ($1 \le w_i \le 1\,000$).

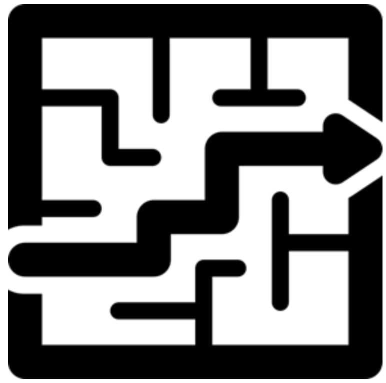
Output

Output n lines. On the ith line, output a single integer: the cost of purchasing tickets if city i wins the contest.

Sample Input 1	Sample Output 1
5 2 5 3 4 1 1 2 2 2 4 5 4 3 3 5 2 6	130 159 191 163 171

Sample Input 2	Sample Output 2
6 4 3 3 4 3 3 1 3 2 2 1 1 1 4 6 4 5 6 6 4 2	209 206 232 209 336 232

Problem F
Get Outta Dodge



Given an orthogonal maze rotated 45 degrees and drawn with forward and backward slash characters (see below), determine the minimum number of walls that need to be removed to ensure it is possible to escape outside of the maze from every region of the (possibly disconnected) maze.

```
/\
\/
```

This maze has only a single region fully enclosed. Removing any wall will connect it to the outside.

```
/\..
\.\.
.\/\
..\/
```

This maze has two fully enclosed regions. Two walls need to be removed to connect all regions to the outside.

Input

The first line has two numbers, R and C, giving the number of rows and columns in the maze's input description. Following this will be R lines each with C characters, consisting only of the characters '/', '\', and '.'. Both R and C are in the range $1 \ldots 1\,000$.

Define an odd (even) square as one where the sum of the x and y coordinates is odd (even). Either all forward slashes will be in the odd squares and all backslashes in the even squares, or vice versa.

Output

Output on a single line an integer indicating how many walls need to be removed so escape is possible from every region in the maze.

Sample Input 1	Sample Output 1
2 2 /\ \/	1

Sample Input 2

```
4 4
/\..
\.\.
.\/\
..\/
```

Sample Output 2

```
2
```

Sample Input 3

```
2 2
\/
/\
```

Sample Output 3

```
0
```

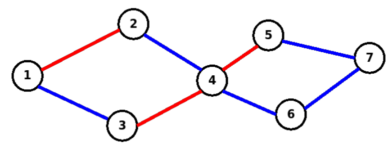
Sample Input 4

```
8 20
/\/\/\/\/\/\/\/\/\/\
\../\.\/./././\/\/\/
/./\.././\/\.\/\/\/\
\/\/\.\/\/./\/..\../
/\/./\/\/./..\/\/..\
\.\.././\.\/\/./\.\/
/.../\../..\/./.../\
\/\/\/\/\/\/\/\/\/\/
```

Sample Output 4

```
26
```

Problem G
On Average They're Purple



Alice and Bob are playing a game on a simple connected graph with N nodes and M edges.

Alice colors each edge in the graph red or blue.

A path is a sequence of edges where each pair of consecutive edges have a node in common. If the first edge in the pair is of a different color than the second edge, then that is a "color change."

After Alice colors the graph, Bob chooses a path that begins at node 1 and ends at node N. He can choose any path on the graph, but he wants to minimize the number of color changes in the path. Alice wants to choose an edge coloring to maximize the number of color changes Bob must make. What is the maximum number of color changes she can force Bob to make, regardless of which path he chooses?

Input

The first line contains two integer values N and M with $2 \leq N \leq 100\,000$ and $1 \leq M \leq 100\,000$. The next M lines contain two integers a_i and b_i indicating an undirected edge between nodes a_i and b_i $(1 \leq a_i, b_i \leq N, a_i \neq b_i)$.

All edges in the graph are unique.

Output

Output the maximum number of color changes Alice can force Bob to make on his route from node 1 to node N.

Sample Input 1	Sample Output 1
3 3 1 3 1 2 2 3	0

Sample Input 2	Sample Output 2
7 8 1 2 1 3 2 4 3 4 4 5 4 6 5 7 6 7	3

Problem H
Pulling Their Weight


Buffalo, vole, and schnauzer. Original images from Wikipedia.

To save money, Santa Claus has started hiring other animals besides reindeer to pull his sleigh via short term 'gig' contracts. As a result, the actual animals that show up to pull his sleigh for any given trip can vary greatly in size.

Last week he had 2 buffalo, 37 voles and a schnauzer. Unfortunately, both buffalo were hitched on the left side and the entire sleigh flipped over in mid-flight due to the weight imbalance.

To prevent such accidents in the future, Santa needs to divide the animals for a given trip into two groups such that the sum of the weights of all animals in one group equals the sum of the weights of all animals in the other. To make the hitching process efficient, Santa is seeking an integer target weight t such that all animals that are lighter than t go in one group and those heavier than t go into the other. If there are multiple such t, he wants the smallest one. There's one small wrinkle: what should be done if there some animals have weight exactly equal to t? Santa solves the problem this way: if there are an even number of such animals, he divides them equally among the two groups (thus distributing the weight evenly). But if there are an odd number of such animals, then one of those animals is sent to work with the elves to make toys (it is not put in either group), and the remaining (now an even number) are divided evenly among the two groups.

Input

Input describes a list of animals' weights. The first line contains an integer m ($2 \le m \le 10^5$), indicating the number of animals. The next m lines each have a positive integer. These are the weights of the animals (in ounces). Animals weighing more than 20 000 ounces are too big to pull the sleigh so no given weight will exceed this maximum.

Output

Output the smallest integer target weight t, as described above. It's guaranteed that it is possible to find such an integer.

Sample Input 1

```
4
3
6
1
2
```

Sample Output 1

```
4
```

Sample Input 2

```
4
11
8
3
10
```

Sample Output 2

```
10
```

Sample Input 3

```
2
99
99
```

Sample Output 3

```
99
```

Problem I
Some Sum

Your friend has secretly picked N consecutive positive integers between 1 and 100, and wants you to guess if their sum is even or odd.

If the sum must be even, output '`Even`'. If the sum must be odd, output '`Odd`'. If the sum could be even or could be odd, output '`Either`'.

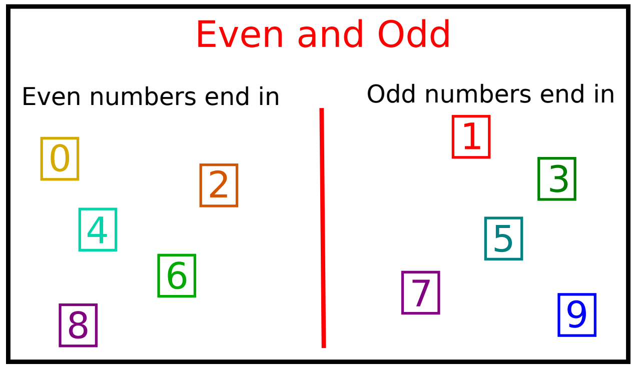


Illustration by Larry Pyeatt

Input

The input is a single integer N with $1 \leq N \leq 10$.

Output

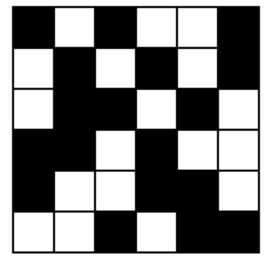
Output a single word. The word should be '`Even`', '`Odd`', or '`Either`', according to the rules given earlier.

Sample Input 1	Sample Output 1
`1`	`Either`

Sample Input 2	Sample Output 2
`2`	`Odd`

This page is intentionally left blank.

Problem J
This Ain't Your Grandpa's Checkerboard



You are given an n-by-n grid where each square is colored either black or white. A grid is *correct* if all of the following conditions are satisfied:

- Every row has the same number of black squares as it has white squares.
- Every column has the same number of black squares as it has white squares.
- No row or column has 3 or more consecutive squares of the same color.

Given a grid, determine whether it is *correct*.

Input

The first line contains an integer n ($2 \leq n \leq 24$; n is even). Each of the next n lines contains a string of length n consisting solely of the characters 'B' and 'W', representing the colors of the grid squares.

Output

If the grid is *correct*, print the number 1 on a single line. Otherwise, print the number 0 on a single line.

Sample Input 1	Sample Output 1
4 WBBW WBWB BWWB BWBW	1

Sample Input 2	Sample Output 2
4 BWWB BWBB WBBW WBWW	0

Sample Input 3	Sample Output 3
6 BWBWWB WBWBWB WBBWBW BBWBWW BWWBBW WWBWBB	0

Sample Input 4	Sample Output 4
6 WWBBWB BBWWBW WBWBWB BWBWBW BWBBWW WBWWBB	1

Problem K
TLNDS

You are given a rooted tree of N vertices where each vertex v_1 through v_N is labeled with a non-negative integer u_1 through u_N. The vertex v_1 is the root.

Define a "jumping path" within the tree to be a sequence $a_1, a_2, \ldots, a_k$ of length k where v_{a_i} is an ancestor of v_{a_j} for all $1 \leq i < j \leq k$.

Compute two quantities for the given tree:

- The length L of the longest jumping path where the labels of the vertices are nondecreasing. That is, $u_{a_i} \leq u_{a_j}$ for all $1 \leq i < j \leq L$.
- The number M of jumping paths of length L where the labels of the vertices are nondecreasing. Since this number may be large, give the remainder of M when divided by the prime $11\,092\,019$.

Input

The first line of input contains an integer N denoting the number of vertices in the tree ($1 \leq N \leq 10^6$).

This is followed by N lines of input indicating the labels u_1 through u_N. Each label is an integer in the range $[0, 10^6]$.

The remaining $N - 1$ lines describe the tree structure. Skipping the root (which has no parent) and starting with $i = 2$, line i gives the parent $p_i < i$ of vertex v_i.

Output

Print a single line of output with two integers separated by a space. The first integer is L, and the second integer is M modulo the prime $11\,092\,019$.

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

Sample Input 2

```
5
4
3
2
1
0
1
2
3
4
```

Sample Output 2

```
1 5
```

Sample Input 3

```
4
1
5
3
6
1
2
3
```

Sample Output 3

```
3 2
```

Sample Input 4

```
6
1
2
3
4
5
6
1
1
1
1
1
```

Sample Output 4

```
2 5
```

Problem L
Weird Flecks, But OK

An artist who wanted to create an installation where his works appeared to be floating in midair has cast a large cube of clear acrylic to serve as a base. Unfortunately, during the casting, some small flecks of dirt got into the mix, and now appear as a cluster of pinpoint flaws in the otherwise clear cube.

He wants to drill out the portion of the cube containing the flaws so that he can plug the removed volume with new, clear acrylic. He would prefer to do this in one drilling step. For stability's sake, the drill must enter the cube perpendicular to one of its faces.

Given the (x, y, z) positions of the flaws, and treating the size of the flaws as negligible, what is the smallest diameter drill bit that can be used to remove the flaws in one operation?

The drill may enter any one of the cube faces, but must be positioned orthogonally to the face.

Input

The first line of input will contain an integer N denoting the number of flaws. $3 \leq N \leq 5\,000$

This is followed by N lines of input, each containing three real numbers in the range $-1\,000.0 \ldots 1\,000.0$, denoting the (x, y, z) coordinates of a single flaw. Each number will contain at most 6 digits following a decimal point. The decimal point may be omitted if all succeeding digits are zero.

Output

Print the diameter of the smallest drill bit that would remove all the flaws.

The answer is considered correct if the absolute or relative error is less than 10^{-4}

Sample Input 1	Sample Output 1
3 1.0 0.0 1.4 -1.0 0.0 -1.4 0.0 1.0 -0.2	2.0000000000

Sample Input 2

```
5
1.4 1.0 0.0
-0.4 -1.0 0.0
-0.1 -0.25 -0.5
-1.2 0.0 0.9
0.2 0.5 0.5
```

Sample Output 2

```
2.0000000000
```

Sample Input 3

```
8
435.249 -494.71 -539.356
455.823 -507.454 -539.257
423.394 -520.682 -538.858
446.507 -501.953 -539.37
434.266 -503.664 -560.631
445.059 -549.71 -537.501
449.65 -506.637 -513.778
456.05 -499.715 -561.329
```

Sample Output 3

```
49.9998293198
```