Problem A. Average

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

In college, a student may take several courses. For each course i, he earns a certain credit c_i, and a mark ranging from `A` to `F`, which is comparable to a score s_i, according to the following conversion table:

A	A-	B+	B	B-	C+	C	C-	D	D-	F
4.0	3.7	3.3	3.0	2.7	2.3	2.0	1.7	1.3	1.0	0

The GPA is the weighted average score of all courses one student may take, if we treat the credit as the weight. In other words,

$$\text{GPA} = \frac{\sum(c_i \cdot s_i)}{\sum c_i}$$

An additional treatment is taken for special cases. Some courses are based on "Pass/Not pass" policy, where students earn a mark '`P`' for "Pass" and a mark '`N`' for "Not pass". Such courses are not supposed to be considered in computation. These special courses must be ignored for computing the correct GPA.

Specially, if a student's credit in GPA computation is 0, his/her GPA will be 0.00.

Input

There are several (about 10) test cases, please process till EOF.

Each test case starts with a line containing one integer N ($1 \leq N \leq 1000$), the number of courses. Then N lines follow, each containing the credit and the mark of one course. Credit is a positive integer less than 10.

Output

For each test case, print the GPA (rounded up to two decimal places; $n \cdot 10^{-2} + 0.005$ is rounded to $(n + 1) \cdot 10^{-2}$) as the answer.

Example

standard input	standard output
5 2 B 3 D- 2 P 1 F 3 A 2 2 P 2 N 6 4 A 3 A 3 A 4 A 3 A 3 A	2.33 0.00 4.00

Problem B. Bobby's Warehouse

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Bobby is a warehouse keeper. He writes down the entry records everyday. The record is shown on a screen, as follows:

```
      Number    2 +
Total prices    5 +
```

There are only two '+' buttons on the screen. Pressing the button in the first line once increases the number on the first line by 1. The cost per unit remains untouched. For the screen above, after the button in the first line is pressed, the screen will be:

```
      Number    3 +
Total prices    7 +
```

The exact total price is 7.5, but on the screen, only the integral part 7 is shown.

Pressing the button in the second line once increases the number on the second line by 1 (so cost per unit is changed accordingly). The number in the first line remains untouched. For the screen above, after the button in the second line is pressed, the screen will be:

```
      Number    3 +
Total prices    8 +
```

Remember the exact total price is 8.5, but on the screen, only the integral part 8 is shown.

A new record will be like the following:

```
      Number    1 +
Total prices    1 +
```

At that moment, the total price is exactly 1.0.

Bobby expects the final screen to be the following:

```
      Number    x +
Total prices    y +
```

Where x and y are given in advance.

What is the minimal number of button presses Bobby needs to achieve his goal?

Input

There are several (about $5 \cdot 10^4$) test cases, please process till EOF.

Each test case contains one line with two integers x ($1 \le x \le 10$) and y ($1 \le y \le 10^9$) separated by a single space — the numbers expected to be shown on the screen at the end.

Output

For each test case, print the minimal number of button presses, or -1 if there is no way to achieve his goal.

Example

standard input	standard output

1 1
3 8
9 31
	0
	5
	11

Note

For the second test case, one way to achieve the goal is:

$(1, 1)$ – $(1, 2)$ – $(2, 4)$ – $(2, 5)$ – $(3, 7.5)$ – $(3, 8.5)$.

Problem C. Campus Design

Input file:	*standard input*
Output file:	*standard output*
Time limit:	6 seconds (12 seconds for Java)
Memory limit:	512 mebibytes

The Eastern University is celebrating its 64-th anniversary. In order to make room for student activities, to make the university a more pleasant place for learning, and to beautify the campus, the college administrator decided to start construction on an open space.

The designers measured the open space and came to the conclusion that it is a rectangle with a length of N meters and a width of M meters. Then they split the open space into $N \times M$ squares. To make it more beautiful, the designer decides to cover the open space with 1×1 bricks and 1×2 bricks, according to the following rules:

1. All the bricks can be placed horizontally or vertically.
2. The vertices of the bricks should be placed on integer lattice points.
3. The number of 1×1 bricks should not be less than C or more than D. The number of 1×2 bricks is unlimited.
4. Some squares have flowerbeds on them. Such squares should not be covered by any brick. (We use 0 to represent a square with a flowerbed and 1 to represent an empty square.)

Now the designers want to know how many ways are there to cover the open space, meeting the above requirements. As this number can be very large, your task is to find the answer modulo $10^9 + 7$.

Input

There are several (about 1000) test cases, please process till EOF.

Each test case starts with a line containing four integers N ($1 \le N \le 100$), M ($1 \le M \le 10$), C, D ($0 \le C \le D \le 20$). Then N lines follow, each being a string with the length of M. The string consists of '0' and '1' only. The j-th character in i-th line is equal to '0' in case if the related square should not be covered by any brick, and '1' otherwise.

Output

Please print one line per test case. Each line should contain an integer representing the answer to the problem modulo $10^9 + 7$.

Example

standard input	standard output

1 1 0 0
1
1 1 1 2
0
1 1 1 2
1
1 2 1 2
11
1 2 0 2
01
1 2 0 2
11
2 2 0 0
10
10
2 2 0 0
01
10
2 2 0 0
11
11
4 5 3 5
11111
11011
10101
11111
	0
	0
	1
	1
	1
	2
	1
	0
	2
	954

Problem D. Destroy the Bases

Input file:	*standard input*
Output file:	*standard output*
Time limit:	12 seconds
Memory limit:	512 mebibytes

At the year of 8192, the war between Evil Army and Galaxy Army broke out. Unfortunately, Evil Army had conquered half the galaxy in just one year. To prevent the situation of the war from getting worse, Levi, the general of Galaxy Elite Army, was ordered by his superior to destroy the enemy's power bases.

Levi was born with the remarkable ability of counter-surveillance, it was just a piece of cake for him to reach Evil Army's power bases. Each power base can be represented as a triangle in 3D-Cartesian coordinate system. The only weapon Levi has is a laser cannon which can shoot in two opposite directions simultaneously. To avoid being caught by enemy, Levi can only place the laser cannon somewhere on a segment from S to T. Unfortunately, there was something wrong with the laser cannon: Levi can not adjust its shooting angle, so the shooting direction is fixed.

Since Levi does not have any time to find the best place to shoot the laser, he decided to select a point on the segment randomly, place the cannon in that point and shoot. If the laser touches a base (even the boundary), that base will be destroyed. Your task is to calculate the expected number of bases destroyed in one shot.

It is recommended to see the sample input to understand the problem statement more clearly.

Input

There are several (about 20) test cases, please process till EOF.

For each test case, the first line is an integer N ($1 \le N \le 10^5$), the number of enemy's power bases. Each of the next three lines contains three integers, x, y and z, denoting the coordinates of S, T, and the fixed shooting direction. Each of the last N lines contain nine integers, x_1, y_1, z_1, x_2, y_2, z_2, x_3, y_3, z_3, denoting the coordinates of the three vertices of enemy's power base. It is guaranteed that none of the triangles will be degenerate, the shooting direction will have at least one non-zero coordinate and the absolute value of all numbers except N and T will not exceed 1000.

Output

For each test case, print the expected number of destroyed power bases. Any answer within an absolute error less than or equal to 10^{-6} will be accepted.

Example

standard input	standard output

2
0 0 0
2 0 0
0 0 1
-1 0 1 1 0 1 -1 0 2
1 1 -1 1 -1 -1 2 0 -1
1
0 0 0
1 0 0
0 1 0
-1 0 0 1 0 0 0 1 0
1
0 0 0
0 0 0
0 1 0
-1 0 1 1 1 1 -1 0 -1
	1.0000000000
	1.00
	1

Problem E. Eastern University Illumination

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

The circular roundabout of 2nd road is an important traffic junction in Eastern University. To celebrate the 64th anniversary, the school officer decides to build some lamps shaped as numbers surrounding the circular roundabout.

There are $2N$ lamps to build. Lamps have the shape of numbers ranging from 1 to $2N$. Each lamp will be built as a whole and can not be split into single digits. For example, lamp 19 can not be split into lamp 1 and lamp 9. Since the roundabout is circular, the last lamp is next to the first lamp. Unfortunately, due to the carelessness of the construction team, some lamps are placed in the wrong position. Since the construction team is now absent after finishing their work, we could only correct the order of lamps by ourselves. After rearrangement, the lamp with the shape i ($1 \leq i \leq 2N$) is expected to be placed on the i-th position. In other words, the final permutation should be $1, 2, \ldots, 2N$. Because of the large volume of the lamps, we have only two ways to adjust the order.

1. Reverse the consecutive four lamps starting from position x, denoted as $(1\ x)$. For example:

   ```
   (1 4): 1 2 3 [4 5 6 7] 8 → 1 2 3 [7 6 5 4] 8
   ```

   or

   ```
   (1 6): 1] 2 3 4 5 [6 7 8 → 6] 2 3 4 5 [1 8 7
   ```

   where '[' stands for the start position, and ']' stands for the end position respectively.

2. Shift all lamps to the left by x positions, denoted as $(2\ x)$. For example:

   ```
   (2 4): 1 2 3 4 5 6 7 8 → 5 6 7 8 1 2 3 4
   ```

   or

   ```
   (2 7): 1 2 3 4 5 6 7 8 → 8 1 2 3 4 5 6 7
   ```

The data guarantees the existence of a valid adjustment sequence. Also the case where all lamps are already placed correctly will not exist in the input. Your task is to generate a valid adjustment sequence that puts all lamps in the correct positions.

Input

There are several (about 150) test cases, please process till EOF.

Each test case starts with a line containing one integer N ($2 \leq N \leq 30$). The next line contains $2N$ different integers, the i-th integer A_i denotes that the shape of the i-th lamp is A_i ($1 \leq A_i \leq 2N$).

Output

For each test case, the first line of the output should contain a single integer $S \leq 4 \cdot 10^5$, denoting the number of swap operations you need to perform.

Each of the following S lines should consist of two integers. The first one may be 1 or 2, indicating the type of swap operations. For the first type, the second number is the start position of the operation, and for the second type, the second number is the number of positions by which you need to shift. In both cases, the second number should be between 1 and $2N$.

If there are multiple possible solutions satisfying the conditions above, any one will be accepted. Please don't print extra empty lines, spaces and other irrelevant characters.

Example

standard input	standard output

4
6 5 4 3 2 1 8 7
4
4 5 1 2 3 6 7 8
4
7 8 4 3 2 1 5 6
	2
	1 2
	1 6
	2
	1 2
	1 1
	2
	1 3
	2 2

Problem F. Fastest Food

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

The campus of the Eastern University can be viewed as a graph with N vertices and M undirected edges (vertices are numbered from 0 to $N-1$). Each edge has the same length 1. Every day, there are K students walking to the dining-hall (vertex $N-1$) from the learning building (vertex 0) at lunch time. They all want reach the dining-hall as soon as possible. However, each edge can only serve at most c_i students at any time. Can you make arrangements for students, so that the last student can reach the dining-hall as soon as possible? (It is assumed that the speed of the students is 1 edge per unit time.)

Input

There are several (about 350) test cases, please process till EOF.

The first line of each test case contains three integers: N ($2 \le N \le 2500$), M ($0 \le M \le 5000$), K ($0 \le K \le 10^9$). Then follow M lines, each line has three integers a_i, b_i, c_i ($0 \le c_i \le 20$), means there is an edge from vertex a_i to b_i with the capacity c_i.

Output

For each test case, print an integer representing the minimum time. If the requirements can not be met, print "`No solution`" (without quotes) instead.

Example

standard input	standard output
`5 6 4`	`3`
`0 1 2`	`6`
`0 3 1`	`No solution`
`1 2 1`	
`2 3 1`	
`1 4 1`	
`3 4 2`	
`3 3 10`	
`0 1 1`	
`1 2 1`	
`0 2 1`	
`2 0 1`	

Problem G. George's Walk

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

George is seriously drunk. He feels as if he is in an N-dimension Euclidean space, wandering aimlessly. In each step, he walks toward some direction and the "length" of each step will not exceed R. Technically speaking, George is initially located at the origin of the N-dimension Euclidean space. Each step can be represented by a random N-dimension vector $(x_1, x_2, \ldots, x_n)$, chosen uniformly from possible positions satisfying $x_i \geq 0$ and ${x_1}^2 + {x_2}^2 + \ldots \leq R^2$.

Assume the expectation of his coordinate after his first step is $(y_1, y_2, \ldots, y_n)$. He wants to know the minimum y_i.

Input

There are several (about 10^4) test cases, please process till EOF.

Each test case consists of only one line containing two integers N and R, representing the dimension of the space and the length limit of each step ($1 \leq n \leq 2 \cdot 10^5,\ R \leq 10^5$).

Output

For each test case, print a real number representing the answer to the question above. Your answer is considered correct if the difference between your answer and the correct one is less than 10^{-6}.

Example

standard input	standard output
2 1	0.4244131816

Problem H. Happy Cirno's Present

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	512 mebibytes

One day, three faeries, Sunny, Luna and Star, visited Cirno's house. Cirno was very happy, so she decided to give a present to them.

Cirno's present was a tree with N nodes. Each node of the tree has an equal possibility to be owned by one of Sunny, Luna or Star. After having decided every node's owner, Cirno cut the edges that connect different owner's nodes, after which each of the three fairies got some connected component of nodes (possibly none).

Then each of the three faeries had to spend some magical energy repairing the components she got. Suppose she had X components containing an odd number of nodes and Y components containing an even number of nodes, the magical energy she would spend equals $\max(0, X - Y)$.

Cirno would compensate for the energy they spent with some food, so she asked you the expectation of the total energy the three fairies would spend, and your task is to find out the answer.

To make it simpler, as it's easy to prove the multiplication of the expectation and the N-th power of three makes an integer, your task is to find the remainder of the multiplication divided by $10^9 + 7$.

Input

There are several (about 10^4) test cases, please process till EOF.

Each test case starts with a line containing one integer N ($1 \le N \le 300$). Then $N - 1$ lines follow, each of them contains two integers u and v ($1 \le u, v \le N$) representing an edge between node u and node v.

Output

For each test case, print the product of the expected energy cost and the N-th power of three, taken modulo $10^9 + 7$, in a line.

Example

standard input	standard output

1
2
1 2
3
2 1
1 3
	3
	12
	51

Problem I. Inspector Fang

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 mebibytes

Famous painter Ka Le Wi loves painting very much. She paints GFW (Great Funny Wall) every day. She has N bags of pigments and is going to be painting for N days. Every day before painting, she produces a wonderful color out of pigments by mixing water and some bags of pigments. On the K-th day, she will select K specific bags of pigments and mix them to get a color of pigments which she will use that day. When she mixes a bag of pigments with color A and a bag of pigments with color B, she will get pigments with color A xor B. When she mixes two bags of pigments with the same color, she will get color zero for some strange reasons.

Her friend, Inspector Fang, has no idea about which K bags of pigments Ka Le Wi will select on the K-th day. He wonders what is the sum of the colors Ka Le Wi will get with $\binom{N}{K} = \frac{N!}{K! \cdot (N-K)!}$ different plans.

For example, assume $N = 3$, $K = 2$ and three bags of pigments with color 2, 1, 2. She can get color 3, 3, 0 with 3 different plans. In this instance, the answer Inspector Fang wants to get on the second day is $3 + 3 + 0 = 6$.

Inspector is so busy that he doesn't want to spend too much time on it. Can you help him? You should tell the answer for each day from the first to the N-th day. As the answers may be large, compute each of them modulo $10^6 + 3$.

Input

There are several (about 250) test cases, please process till EOF.

For each test case, the first line contains a single integer N ($1 \le N \le 10^3$). The second line contains N integers a_i ($0 \le a_i \le 10^9$). The i-th integer a_i represents the color of the pigments in the i-th bag.

Output

For each test case, output N integers in a line representing the answers (modulo $10^6 + 3$) from the first day to the N-th day.

Example

standard input	standard output

`4`
`1 2 10 1`
	`14 36 30 8`

Problem J. John's Game

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

John likes metal balls. He has some balls and he wants to arrange them in a row on the table. Each of those balls can be one of three possible colors: red, yellow, or blue. More precisely, John has R red balls, Y yellow balls and B blue balls. He may put these balls in any order on the table, one after another. Each time John places a new ball on the table, he may insert it somewhere in the middle (or at one of the ends) of the already-placed row of balls.

Additionally, each time John places a ball on the table, he scores some points (possibly zero). The number of points is calculated as follows:

- For the first ball being placed on the table, he scores 0 points.
- If he places the ball at one end of the row, the number of points he scores equals to the number of different colors of the already-placed balls (that is, all balls except the new one) on the table.
- If he places the ball between two balls, the number of points he scores equals to the number of different colors of the balls before the currently placed ball, plus the number of different colors of the balls after the current one.

What's the maximal total number of points that John can earn by placing the balls on the table?

Input

There are several (about 1500) test cases, please process till EOF.

Each test case contains only one line with 3 integers R, Y and B, separated by single spaces. All numbers in input are non-negative and won't exceed 10^9.

Output

For each test case, print the answer in one line.

Example

standard input	standard output

2 2 2
3 3 3
4 4 4
	15
	33
	51

Problem K. King's Tree

Input file:	*standard input*
Output file:	*standard output*
Time limit:	6 seconds (15 seconds for Java)
Memory limit:	512 mebibytes

There is a skyscraping tree standing near the King's palace. On each branch of the tree is an integer. (The tree can be treated as a connected graph with N vertices, while each branch can be treated as a vertex). Today the King's advisors under the tree are considering a problem: can we find such a chain on the tree so that the product of all integers on the chain taken modulo $10^6 + 3$ equals to K? Can you help them in solving this problem? A chain on a tree is a path between two different vertices of the tree which does not contain any vertex twice.

Input

There are several (about 1000) test cases, please process till EOF.

Each test case starts with a line containing two integers N $(1 \le N \le 10^5)$ and K $(0 \le K < 10^6 + 3)$. The following line contains n numbers v_i $(1 \le v_i < 10^6 + 3)$, where v_i indicates the integer on vertex i. Then follow $N - 1$ lines. Each line contains two integers x and y representing an undirected edge between vertex x and vertex y.

Output

For each test case, print a single line containing two integers a and b (where $a < b$), representing the two endpoints of the chain. If multiple solutions exist, please print the lexicographically smallest one (first minimize a, then minimize b, if needed). In case no solution exists, print "`No solution`" (without quotes) instead.

For more information, please refer to the example below.

Example

standard input	standard output

5 60
2 5 2 3 3
1 2
1 3
2 4
2 5
5 2
2 5 2 3 3
1 2
1 3
2 4
2 5
	3 4
	No solution