A - Abnormal 95ers

Once again we have the first problem about abnormal newcomers of Amirkabir University of Technology. They are called 95ers and they are abnormal. You wonder why they are called abnormal? I'll tell you why. They want to organize a competition to find which team is the best among all football teams. As you know, most of tournaments in the world follow this rule that teams are partitioned into groups of four and then the first two teams of each group are qualified for the next round. But these students want to form groups of size three while only the best team of each group will proceed to the next round.

This is how they are doing it. First they find N teams (N is a power of 3, like 27, 81, etc.) then they put these teams in groups of size three. Then the competition begins and each team plays once against other teams in its group and the best team of each group qualifies for the next round. From the qualifiers, groups of size 3 are formed and in these groups teams play against each other, and this process continues until only one team remains and that's the champion.

For example if initially there are 27 teams, in the first round they form 9 groups of size 3, then in the second round 9 qualified teams form 3 groups of size 3, then in round three (the last round) 3 teams that qualified from these 3 groups create one group of size 3 and that's where the champion is identified. 27 games are played in the first round, 9 games are played in the second round, 3 games are played in the third round which is 39 games totally.

They want to reserve the football field so they have to figure out how many games will be played between the teams? Given the number of teams can you figure out the number of games that is going to be played?

Input Format

The first and only line of input contains an integer N, the number of teams.

Constraints

$1 \leq N \leq 400{,}000{,}000$, and N is a power of 3.

Output Format

Print one line in the output which contains a single integer, the total number of games that are going to take place in this competition.

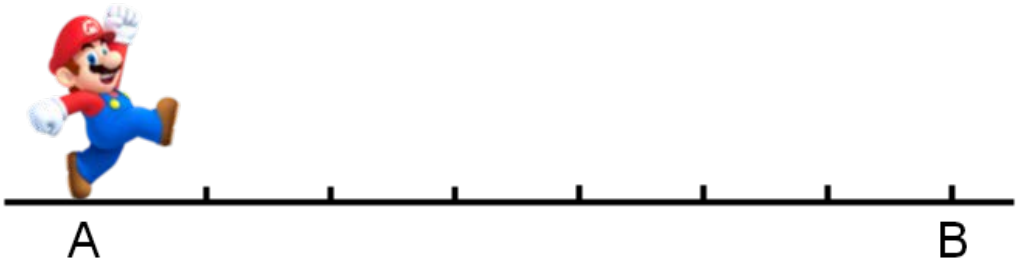
Sample Input	Sample Output
3	3
27	39

B - Super Mario

Do you remember Super Mario? He was always looking for mushrooms to become bigger and more powerful. Now he is standing at point A and wants to go to point B to eat some mushrooms. In order to walk less, he can use some teleports. Teleports are designed in such a way that Super Mario can enter any of them and get out of any other one that he likes and it has no cost for him. There are teleport devices at every locations P where P is a prime number. A prime number is an integer greater than 1 that has no positive divisors other than 1 and itself, e.g. 2, 3, 5, 7, etc. Super Mario may not to use any teleport at all and when he is walking toward the other point he can walk over a teleport and not use that.

For example when Super Mario wants to go from point 10 to point 18 he can walk to point 11, then use that teleport and get out of 17 and then walk from 17 to 18, that is a total of 2 meters.

You have to calculate the minimum distance Super Mario has to walk to get from point A to point B.



Input Format

The first and only line of input contains two positive integers A and B, the starting point of Super Mario and where he wants to get, respectively.

Constraints

$1 \leq A, B \leq 5000$

Output Format

Print one line of output that contains the minimum distance Super Mario has to walk to get from A to B.

Sample Input	Sample Output
3 21	2
32 16	2

C - Contest Security Team

CST stands for "Contest Security Team", it is a team that organizes a programming contest and supervises the preparation process. They have designed a system that helps judges and staffs communicate with each other in a secure way, but how do they do that? They have 3 different encryption methods that encrypts the messages and sends them to the other parties. Each sentence is encrypted using only one of the methods.

Here are the encryption algorithms:

- Random Shuffle Method:
  - Just shuffle randomly the characters in every word of the sentence, while preserving the order of the words.
  - For example word "beaUtiful1367" can be replaced with "67Ulabefuti13"

- Rotate Method:
  - Replace each character with its next character (case-sensitive and circular within the three blocks 0-9, A-Z and a-z), (e.g. replace a with b, replace m with n, … and replace z with a, and the same process for uppercase letters, and replace 0 with 1, 1 with 2, ... and 9 with 0)
  - For example word "Love1988" is replaced with “Mpwf2099”

- Minimum Neighbor Method
  - In one go, replace each character with minimum of its left character, itself and its right character. The first and the last characters remain intact. Character comparison is based on the ASCII values (0...9 < A...Z < a...z), (e.g. minimum(a, Z, 0) = 0, minimum(b, c, a) = a).
  - For example word “AmirKabir2016” is replaced with “AAiKKKab20006”.

Judges’ encryption system has some issues at the moment and they can’t decrypt the incoming messages, hence they can’t understand what staffs are saying. Given the sentence, can you find which method has been used to encrypt it?

A *sentence* is a sequence of one or more *word*s separated with a single space and is **terminated with a period.**
A *word* is a sequence of alphanumeric characters (0-9, A-Z, a-z).

Input Format

The first line of input contains an integer N, the number of all of the words that are possible to be used in the communications between the contest organizers.
The following N lines each contains a *word*.
The last line contains an encrypted *sentence*.

Constraints

$1 \leq N \leq 100$
Each sentence is composed of at most 20 words.
Each word is a non-empty string composed of at most 20 alphanumeric characters.

Output Format

Print one line of output that contains the method that has been used to encrypt the input sentence. Print “Random Shuffle”, “Rotate” or “Minimum Neighbor” in the output, or print “None” if the sentence cannot be decrypted using any of the above methods. It is guaranteed that the sentence can be decrypted with **at most** one of the methods.

Sample Input	Sample Output
7 16th Amirkabir International Programming Contest November 2016 otCnste mirariAkb stCotne krbaAirim h6t1 61ht.	Random Shuffle
7 16th Amirkabir International Programming Contest November 2016 Dpouftu 27ui 3127 Qsphsbnnjoh Dpouftu Dpouftu.	Rotate

D - Moving To Cuba

People are scared of what is going to happen next if David Trueman becomes the next President, so they have rushed to the seaport to move to Cuba. Now they are waiting to get in the ship. People are getting on board one by one. It takes one minute to get a person on board. Their tempers are different, meaning they get angry with different rates. For example if the initial level of angriness of someone is 5 and he gets angry with a rate of 3, it means that if he/she stands in the line for zero minutes then he/she will get into the ship with a level of angriness of 5, if he/she stands in the line for one minute then he/she will get into the ship with a level of angriness of 8, two minutes in the line and he/she will get into the ship with a level of angriness of 11 and so on.

We want to reorder the people in such a way that the level of angriness of the angriest person that gets on the ship becomes as low as possible. Given the initial level of angriness of the people and the rates in which their angriness increases for each minute they wait in the line. Find the level of angriness of the angriest person that gets in to the ship if we do our best.

Input Format

The first line of input contains an integer N which is the number of people who want to get into the ship to move to Cuba. Each of the next N lines contains 2 integers A_i and B_i, the initial angriness of a person and the rate in which the angriness of the person increases.

Constraints

$1 \leq N \leq 100{,}000$, $0 \leq A_i, B_i \leq 1{,}000{,}000$

Output Format

Print one line of output that contains the maximum level of angriness that gets into the ship if we reorder the people in the best possible way.

Sample Input	Sample Output
2 2 1 3 2	3
4 5 5 2 3 4 5 2 4	11

E - The Matrix

You are given a matrix of size N×N. Each cell of the matrix contains one English letter (uppercase or lowercase). Cells of the matrix are connected together in four directions, up, down, left and right. A component is the maximal subset of cells with the same character connected to each other. Size of a component is the number of cells in that component.

For example:

bold B's form a component of size 7	bold B's doesn't form a component
C**B**CC **BB**CC **BBB**b C**B**CB	C**B**CC **BB**CC **BBB**b CBCB

Now you can change at most one cell of the matrix to any character you like. You have to do this in a way to maximize the size of the largest component.

Input Format

First line of input contains an integer N, the number of rows and columns of the matrix.
The next N lines each contains N characters, representing the matrix.

Constraints

$1 \leq N \leq 500$

Output Format

Print one line of output that contains an integer, the size of the largest component after you change that cell.

Sample Input	Sample Output
5 abcda aabca abbca aabba aabaa	15

6 bcdace cbadcc eeabcd ceaddb ccadbb ccbdcd	7

F - AUT Land

Amirkabir University of Technology, formerly called Tehran Polytechnic, referred to as "The Mother of Engineering Universities" in Iran, has more than 15 departments. The main campus of Amirkabir University of Technology is in Tehran, Iran. It is located close to Vali-Asr crossroads, the intersection of Enghelab Street and Vali-Asr Street, in the very center of Tehran City. Many students commute to AUT via the subway by Theatre Shahr station.

The university was founded by Habib Nafisi in 1956 and developed by Dr. Mohammad Ali Mojtahedi, during the reign of the Pahlavi dynasty. Named the Tehran Polytechnic, it began with five engineering departments. Six months before the victory of 1979 Iranian Revolution, Tehran Polytechnic was renamed after the Iranian Prime minister Amir Kabir (1807–1852).

Now that you have a good understanding of the host of this contest, let's get straight to the problem statement. We have a tree G of size N (in the graph theory not in a garden!), we want to add a new node v to G in order to have another tree F, i.e. $V_F = V_G \cup \{v\}$ and $E_F = E_G \cup \{(u, v)\}$ where $u \in V_G$. Two ways of adding node are different if we select different neighbor for v (i.e. different u in the above formula). So we can add a new node in N different ways.

We want to know what is the maximum number of diameters after adding a new node, and in how many ways we can add a new node to have the maximum number of diameters?

*"In mathematics, and more specifically in graph theory, a **tree** is an undirected graph in which any two vertices are connected by exactly one path. In other words, any connected graph without simple cycles is a tree."* Wiki
*"The length of the "longest shortest path" (i.e., the longest graph geodesic) between any two graph vertices of a graph is called graph **diameter**."* Wiki

Input Format

The first line of the input contains a single integer n, the number of nodes in the tree. Next line contains $n-1$ positive integers $p_2, p_3, p_4, \ldots, p_n$ $(1 \le p_i \le i-1$ for $2 \le i \le n)$ — the description of the edges in the tree. Number p_i means that the tree has an edge connecting node p_i and node i.

Constraints

$2 \le n \le 100{,}000$

Output Format

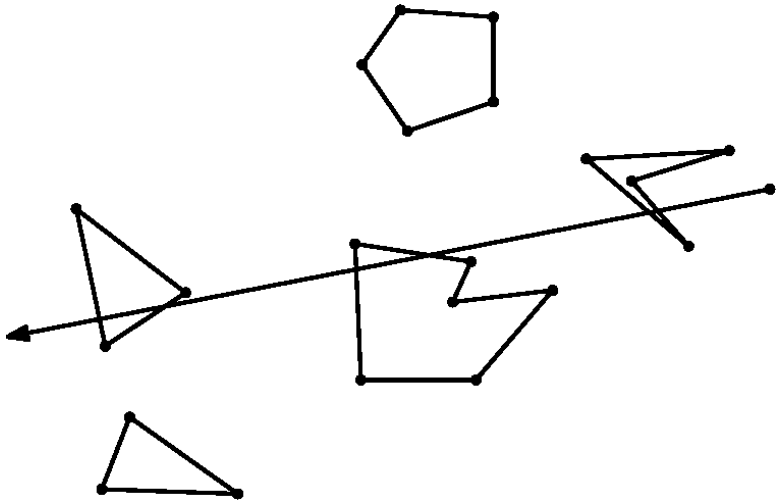
Print one line of output containing two integers separated by a single space, the number of ways we can add a new node to maximize the number of diameters in the resulting graph and the number of diameters after we added that new node, respectively.

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

G - Shooting For The Stars

Poopi and MeHdi live in a 2D world. They want to shoot a laser beam and blow up as many stars as they can with their laser. They can shoot the laser beam from any point in any direction. A star is blown up if the laser touches it even in a single point.

Each star is like a simple polygon in their world. Given the position of the stars, can you find the maximum number of stars they can blow up with their laser? Laser beam passes through the stars and is not obscured or reflected by them.



Input Format

First line of input contains an integer N, the number of stars.

Each star is described as a simple polygon in the following format:

S

$x_1\ y_1$

$x_2\ y_2$

...

$x_S\ y_S$

S is the number of polygon's vertices and it follows with S points, the position of the vertices.

For more details look at the sample input.

Constraints

$1 \leq N \leq 67$

$3 \leq S_i \leq 6$

$-2000 \leq x_j, y_j \leq 2000$

It is guaranteed that all points are different.

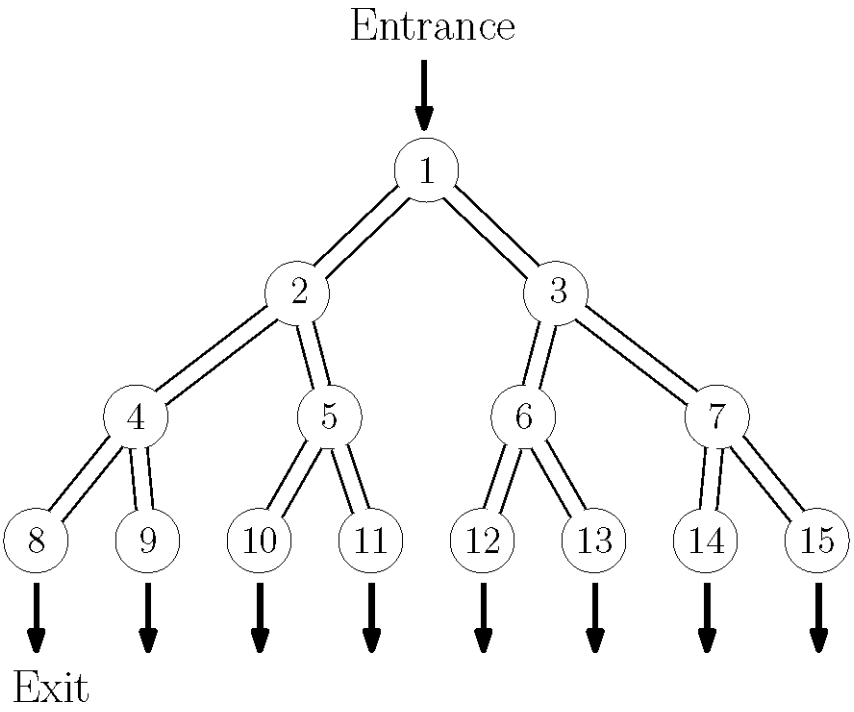
Output Format

Print one line of output that contains an integer M, the maximum number of stars they can blow up.

Sample Input	Sample Output
4 4 5 1 7 4 8 5 6 6 3 1 0 1 1 2 0 4 1 -10 4 -15 3 -10 2 -15 3 -1 1 -3 1 -2 2	3

H - Gold Miners

MeHdi needs at least G kilograms of gold for his new project, hence he has bought a gold mine. This mine, just like any other binary mine, is formed of caves that are like full binary trees, in such a way that caves are edges of the tree and the intersection of the caves are nodes of the tree. The entrance of the mine is the root of the tree and leaves of the tree are exits of the mine (like the following picture).



MeHdi wants to enter the mine and exit with at least G kilograms of gold. He can't turn back in the middle of his path, he can only go forward to the exits. As a result, each time he arrives at an intersection he has two ways to continue (the left cave and the right cave). He also knows there are some gold at each intersection, but he knows nothing about distribution of golds in the mine. So in each intersection he will collect all the golds in that intersection and then randomly chooses which cave to follow with equal probability.

Amir as The Lord of The Mines knows exactly how much gold there are in each intersection. Also he is able to destruct K caves so that they are blocked and MeHdi cannot use them to go to the next intersection. If MeHdi arrives at an intersection that both of its caves are blocked he will get trapped there forever :(and if one of its caves is blocked then he will use the other cave of that intersection.

Amir knows that MeHdi is a good ICPC judge so he wants to destruct at most K of the caves so that the probability of gathering at least G kilograms of gold by MeHdi becomes maximum. But Amir is busy coordinating the contest, so he needs you to calculate what would be the maximum probability of collecting at least G kilograms of gold by MeHdi if Amir does his best. He gives you the information of the mine and the number K. If you can calculate this value correctly, in addition to a balloon you will be given a pack of cookies :).

Input Format

First line of input contains four integers, N, M, K and G, the number of nodes, the number of nodes which contains gold, the number of edges you can cut and G as described in the statement, respectively.

Each of the next M lines contains two integers x_i y_i, which means there are y_i kilograms of gold on node x_i (Nodes are numbered from top to bottom and from left to right. See the above picture).

Constraints

$1 \leq M \leq N \leq 100{,}000$

$0 \leq K < N$

$1 \leq G \leq 100{,}000$

$1 \leq x_i \leq n$

$0 \leq y_i \leq 100{,}000$

It is guaranteed that there is an integer p such that $N = 2^p - 1$.

Output Format

Print output in format p/q such that $(p, q) = 1$.

See the sample output for more details.

Sample Input	Sample Output
63 20 2 13 1 1 2 2 3 3 4 4 5 5 6 1 7 2 8 3 9 4 10 5 11 1 12 2 13 3 14 4 15 5 32 1 33 2 34 3 35 4 36 5	11/16

I - DUZip

DUZip is a method for compressing sequences of characters. It consists of two steps.

First step takes two strings as input and generates a sequence of integers as output. Inputs are:

1. *T*: a plain text
2. *P*: a permutation of characters that are used in *T*.

For example, if *T* is "ccaabccbcabcc" then there are six possible values for *P*: {abc, acb, bac, bca, cab, cba}.

In this step we iterate on *T* from left to right and for each character C, we do the following:

1. Append to output index of C in *P*,
2. Remove C from *P* and insert it again at the beginning of *P*.

As another example, if *T* is "cccabcc" and *P* is "abc", then the steps of algorithm are:

***T*[i]**	***P***	**Output**	**Updated *P***
c	abc	[2]	cab
c	cab	[2,0]	cab
c	cab	[2,0,0]	cab
a	cab	[2,0,0,1]	acb
b	acb	[2,0,0,1,2]	bac
c	bac	[2,0,0,1,2,2]	cba
c	cba	[2,0,0,1,2,2,0]	cba

Output of the second step is a compression of the output from the previous step where each maximal consecutive subsequence of identical integers x is replaced with two numbers, x itself and length of that subsequence. Here is a formal definition for the second step as a function:

$F([\,]) = [\,]$
$F(A) = [A[0], run(A)] + F(suffix(A, run(A)))$

Where:

- [ ] is an empty sequence
- $A[0]$ = first element of A
- $run(A)$ = Maximum R such that for all $i = 0..R-1$ we have $A[i] = A[0]$.

- $suffix(A, i)$ = suffix of A starting at index i.

For example, if the input of second step is [2, 0, 0, 1, 2, 2, 0], then its output is [2, 1, 0, 2, 1, 1, 2, 2, 0, 1].

For a given input *T*, we can choose different values for *P* as the input of first step, which may impact the length of output of second step. Length of the final output is the total number of integers in the sequence. Write a program that calculates the minimum length of output for all possible *P*s, and the number of different *P*s that generate an output with minimum length.

Input Format

First and only line of input contains a string *T*.

Constraints

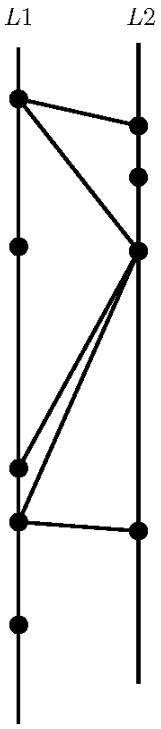
$1 \leq |T| \leq 30{,}000$
T is a string composed of alphanumeric characters.

Output Format

Print one line of output containing two integers, the minimum possible length of DUZip output and the number of different values for *P* that generate a DUZip output with minimum length, respectively. Print the numbers modulo 1,000,000,007.

Sample Input	Sample Output
1	2 1
abccba	8 1
ABBCDEBAF	12 10

J - Game Of Poopi And MeHdi



Poopi and MeHdi just invented a new game. There are two parallel lines L1 and L2 and there are m and n points on them, respectively. Each move consists of connecting one of the points on L1 to one of the points on L2 with a straight segment in such a way that it doesn't cross any of the previous segments except in the end points.

Poopi starts the game and the player who cannot make a move according to the rule loses the game. Given the points on L1 and L2, can you figure out who is the winner assuming that both players play optimally?

Input Format

First line of input contains two positive integers m and n, the number of points on L1 and the number of points on L2, respectively.
Then follows m lines containing points of L1.
Then follows n lines containing points on L2.
For more details look at the sample input.

Constraints

$1 \leq m, n \leq 400$
$0 \leq x_i, y_i \leq 5000$
It is guaranteed that L1 and L2 are two different vertical lines and no two points coincide.

Output Format

If Poopi wins the game print “Poopi” and if MeHdi wins print “MeHdi”.

Sample Input	Sample Output
6 3 1 17 1 8 1 67 1 1 1 3 1 2 5 8 5 11 5 88	MeHdi

K - The Circus

Khalil Oghab is the owner of a famous circus in Tehran. In this circus there is only one circle of chairs that are numbered from 1 to n which means each chair has exactly two other chairs next to it. So chair 1 is next to chairs n and 2, chair 2 is next to chairs 1 and 3, etc.

All of the n chairs are reserved for tonight and everyone knows which chair he/she has to sit on. When someone goes to the circus, if he/she is left-handed he/she puts his/her left hand on the left armrest of his/her chair and if he/she is right-handed he/she puts his/her right hand on the right armrest of his/her chair. The problem is that between any two chairs there is only one armrest and it is possible that two people want to put their hands on the same armrest which is a collision.

Khalil doesn't know people who have reserved the chairs, hence he has no idea who is left-handed and who is right-handed. He wants to know that how many states with exactly k collisions exist. Two states A and B are different if there is one chair that in state A a left-handed person sits on it and in state B a right-handed person sits on it, or vice versa. For example if 5 tickets have been sold, there are 32 different states.

Input Format

First and only line of input contains two integers n and k as described in the problem statement.

Constraints

$3 \leq n \leq 100{,}000$
$0 \leq k \leq n$

Output Format

Print one line in the output which contains a single integer, the number of different states with exactly k collisions. Print the answer modulo 1,000,000,007.

Sample Input	Sample Output
3 1	6