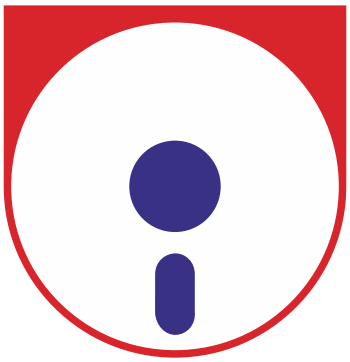


Croatian Open Competition in Informatics

Round 1, October 19th 2019

Tasks

Task	Time limit	Memory limit	Score
Trol	1 second	512 MiB	50
Lutrija	2 seconds	512 MiB	70
Džumbus	1 second	512 MiB	110
Trobojnica	2 seconds	512 MiB	110
Zoo	2 seconds	512 MiB	110
Total			450

Task Trol

Stjepan recently received his bachelor's degree in mathematics from the University of Zagreb. Naturally, his parents are very proud and have decided to give him all positive integers not greater than 2^{60} as a gift. To keep them safe, he quickly stored all of those numbers in an array A, such that $A_i = i$.

His jealous friend Marin decided to prank him by repeatedly replacing each element of A with the sum of its digits until all elements of A consisted of a single digit. For example, the initial value of 197^{th} element of A was 197. Marin first changed that value to $1 + 9 + 7 = 18$ and then changed its value again to $1 + 8 = 9$.

Stjepan is devastated and begs Marin to return his array to its initial state. Unfortunately, Marin won't do that until Stjepan correctly answers his Q queries: "What is the sum of numbers from l-th to r-th element of A".

Help Stjepan answer those queries!

Input

The first line contains an integer Q ($1 \leq Q \leq 100$) from the task description.
The next Q lines contain two integers l_i i r_i ($1 \leq l_i \leq r_i \leq 2^{60}$), the parameters of Marin's i-th query.

Output

Output the answers to each of Marin's Q queries. Each answer should be printed in a separate line and their order should match the order of the queries as they are given in the input.

Scoring

In test cases worth a total of 10 points, for each query will hold $1 \leq l_i \leq r_i \leq 9$.
In test cases worth a total of 30 points, for each query will hold $r_i - l_i \leq 1000$.

Examples

input	**input**	**input**
1 1 5	2 9 13 44 45	1 1998 2018
output	**output**	**output**
15	19 17	102

Clarification of the second example:

1st query → $A_9 = 9$, $A_{10} = 1 + 0 = 1$, $A_{11} = 1 + 1 = 2$, $A_{12} = 1 + 2 = 3$, $A_{13} = 1 + 3 = 4$.
$A_9 + A_{10} + A_{11} + A_{12} + A_{13} = 9 + 1 + 2 + 3 + 4 = 19$.

2nd query → $A_{44} = 4 + 4 = 8$, $A_{45} = 4 + 5 = 9$. $A_{44} + A_{45} = 8 + 9 = 17$.

Task Lutrija

Grandpa Vedran is watching his favorite lottery show on TV in the hopes of becoming an overnight millionaire. The lottery balls are spinning and bouncing around before yielding the following draw: 2, 5, 7, 11, 19, 23 and 31.

Vedran sighs as he didn't guess a single one of those numbers. "Looks like I'm passed my prime...", he thought to himself while turning off the old TV. His vision is also getting worse, so he pressed the wrong button on the remote control and switched to the COCI channel.

The host, Mr. Malnar, calmly spoke: "Dear viewers, on the left side of the screen I will show you a prime number A and on the right side of the screen I will show you a prime number B. The first person that calls in with an array of prime numbers which starts with A, ends with B and has a prime absolute difference between each two neighbouring elements will receive a free trip to IOI 2020 in Singapore."

Old Vedran is reminiscing about his glory days of being a competitive programmer. Unfortunately, he is rusty and is not able to solve the problem. Being kindhearted, you decide to help Vedran win a trip to Singapore.

Note: A prime number is a positive integer greater than 1 that is only divisible by 1 and itself.

Input

The first line contains two prime numbers A and B ($2 \leq A, B \leq 10^{14}, A \neq B$) from the task description.

Output

If the task is impossible, i.e., there is no array satisfying the conditions from task statement, simply output `-1` in a single line.

Otherwise, in the first line output the number of elements in the array and in the second line output its elements separated by spaces. The size of array must not be greater than 30 and its elements must not be greater than 10^{15}. It is guaranteed that, if a solution exists, there is at least one satisfying those bounds.

If there are multiple correct solutions, output any of them.

Scoring

In test cases worth a total of 14 points, it will hold that if a solution exists, there is at least one such that the number of elements in the resulting array is not greater than 3 and all of its elements are not greater than 1000.

In test cases worth additional 28 points, it will hold $2 \leq A, B \leq 1000$.

Examples

input	**input**	**input**
`13 11`	`37 11`	`2 17`
output	**output**	**output**
`2` `13 11`	`-1`	`3` `2 19 17`

Task Džumbus

Marin is a good man, so he'll organize Q parties for his N friends, all of which are competitive programmers. The only drink that is going to be served at his parties will be *džumbus* — a mixture of Coke and ginger juice.

For each of his friends, Marin knows the amount of džumbus they need to drink in order to relax. He also knows that there are M pairs of people among his friends such that, if both of them are relaxed, they will begin to exchange the solutions of past COCI problems (since there are no published editorials). When a person A shares their solutions with person B, the person B may decide to share those solutions in the same manner, but it is also known that M pairs are formed in a way that it is impossible that those solutions will get back to person A during that party, regardless of the order in which exchanges took place.

Marin has prepared different amounts of džumbus for each party. During each party, he will distribute the drink in a way which maximizes the number of people that will at least once exchange their solutions with another person at that party.

Your task is to determine the number of people that will exchange their solutions for each of the Q parties.

Input

The first line contains integers N and M from the task description.
The second line contains N space separated integers D_i, the amounts of džumbus needed to relax Marin's friends, given in order from a friend number 1 to a friend number N.
The i-th of the next M lines contains two integers A_i and B_i ($A_i \neq B_i$), denoting a pair of friends from the task description.
The next line contains an integer Q from the task description.
The next Q lines contain a single integer S_i which represents the total amount of džumbus that is going to be served at i-th party.

Output

Output the number of people that will exchange their solutions for each of the Q parties. The answer for each party should be given in a separate line. Note that the parties are independent of each other.

Scoring

In all subtasks, it holds $0 \leq M < N \leq 1000$, $1 \leq Q \leq 2 \cdot 10^5$, $1 \leq D_i \leq 10^9$ and $1 \leq S_i \leq 10^9$.

Subtask	Score	Constraints
1	20	$M = N - 1$, $1 \leq S_i \leq 1000$, Marin's friends will be paired up in a way that each friend will exchange their solutions with at most two other people.
2	30	$M = N - 1$ Marin's friends will be paired up in a way that each friend will exchange their solutions with at most two other people.
3	30	$N \leq 100$
4	30	No additional constraints.

Examples

input

```
1 0
1000
1
1000
```

output

```
0
```

input

```
3 2
1 2 3
1 2
1 3
3
2
3
5
```

output

```
0
2
2
```

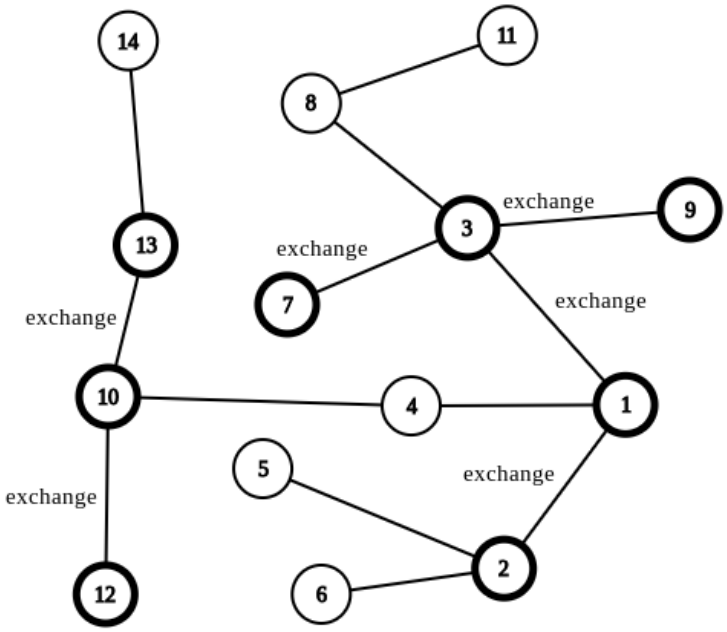
input

```
14 13
2 3 4 19 20 21 5 22 6 7 23 8 10 14
1 2
1 3
1 4
2 5
2 6
3 7
3 8
3 9
4 10
8 11
10 13
10 12
12 14
3
45
44
23
```

output

```
8
7
5
```

Clarification of the third example: At the first party, Marin decided to relax friends with indexes 1, 2, 3, 7, 9, 10, 12 and 13. They have drunk a total of 45 units of džumbus.



Task Trobojnica

„*Everything will be in flames once red, white and blue start running through your veins*" – Slaven Bilić

In this task, we will observe regular polygons that have each of their N sides colored in one of three colors and whose vertices are denoted from 1 to N in a clockwise order. A *triangulaton* of a polygon is a decomposition of its area into a set of non-overlapping triangles whose sides either lie on the sides of the polygon or its internal diagonals. Of course, in this task, each of the diagonals used for polygon triangulation is also colored in one of three colors.

The triangulation is said to be *patriotic* if each of its $N-2$ triangles has all three sides of different colors. Your task is to determine a patriotic triangulation of a given polygon.

Input

The first line contains an integer N from the task description.
The second line contains an integer consisting of N digits which represent the colors of polygon sides. More precisely, the first digit represents the color of side $(1, 2)$, the second digit represents the color of side $(2, 3)$, and so on until the N-th digit which represents the color of side $(N, 1)$. The colors will always be denoted with digits 1, 2 and 3.

Output

If there is no patriotic triangulation of the given polygon, output `NE` and terminate the program. Otherwise, in the first line output `DA` and in the next $N-3$ lines output each diagonal used in the patriotic triangulation. Each diagonal should be specified as X Y C where X and Y are the endpoints of a diagonal and C is its color ($1 \leq X, Y \leq N, 1 \leq C \leq 3$). The order of diagonals in the output is not important. If there are multiple correct solutions, output any of them.

Scoring

Subtask	Score	Constraints
1	20	$4 \leq N \leq 11$
2	40	$4 \leq N \leq 10^3$
3	50	$4 \leq N \leq 2 \cdot 10^5$

If your program correctly outputs the first line in each test case of a certain subtask, you will score 10% of the points allocated for that subtask.

Examples

input

```
4
1212
```

output

```
DA
1 3 3
```

input

```
4
1213
```

output

```
NE
```

input

```
7
1223121
```

output

```
DA
1 3 3
3 5 1
5 7 3
7 3 2
```

Task Zoo

On a cold Christmas night in 2010, Zagreb was covered in snow. Zdravko decided to leave his castle, cross the road and take a stroll through the Maksimir park. Unfortunately, the idyllic winter evening was interrupted by a monster that was lurking in the nearby bushes. The monster jumped in front of Zdravko, but Zdravko roared a mighty roar which scared the monster away. Unbeknownst to him, a bunch of animals from the nearby zoo were startled by that roar. More precisely, a group of tigers and bulls were so annoyed that they decided to escape the zoo in order to find a nice quiet place to sleep.

During their escape, the animals had to pass through a rectangular area divided in $R \times C$ unit squares. The animals must enter the area through the upper-left corner and must leave the area through the lower-right corner. In order to escape as quietly as possible, animals will pass through this area one by one, walking along an arbitrary path in four general directions (up, down, left and right). In other words, each animal does not necessarily travel along a shortest path during its escape and it might step on the same unit square more than once (including the entrance and exit). Since the ground is covered in snow, animals leave footprints when they step inside unit squares. If another footprint is already in the square that is about to be stepped on, the animal will erase the previous footprint and leave a new one.

Your task is to determine the minimal number of animals that have escaped the Maksimir zoo based on the footprints that were left in the aforementioned rectangular area.

Input

The first line contains two integers R and C from the task description.
The next R lines contain C characters that represent the rectangular area from the task description. Character `T` represents a tiger's footprint, character `B` represents a bull's footprint and character `*` represents untouched snow.

You can assume that the input data is such that at least one animal entered the rectangular area and that each animal that entered the area has also left the area according to the rules from the task description.

Output

Output the minimal number of animals that have escaped the zoo.

Scoring

Subtask	Score	Constraints
1	45	$2 \leq R, C \leq 100$
2	65	$2 \leq R, C \leq 1000$

Examples

input

```
4 4
TT*T
*TTT
***T
***T
```

output

```
1
```

input

```
3 5
TTBB*
*T*B*
*TTTT
```

output

```
2
```

input

```
7 5
BT***
BTBBB
BTTTB
BBT*B
BBT*B
BBT**
*BBBB
```

output

```
3
```

Clarification of the second example: