

KTH Challenge 2022

May 7, 2022



Problems

- A Name Generation
- B Patrick's Triangle
- C 99 Problems
- D Box and Arrow Diagram
- E Emergency Exit
- F Efficient Bus Routing
- G The Great Egg Hunt
- H Bar Classification
- I Sperhling

Do not open before the contest has started.

This page is intentionally left (almost) blank.

Advice, hints, and general information

- The problems are not sorted by difficulty.
- If you think some problem is ambiguous or underspecified, you may ask the judges for a clarification request through the Kattis system. The most likely response is “No comment, read problem statement”, indicating that the answer can be deduced by carefully reading the problem statement or by checking the sample test cases given in the problem.
- Your submissions will be run multiple times, on several different input files. If your submission is incorrect, the error message you get will be the error exhibited on the first input file on which you failed. E.g., if your instance is prone to crash but also incorrect, your submission may be judged as either “wrong answer” or “run time error”, depending on which is discovered first.

This page is intentionally left (almost) blank.

Problem A

Name Generation

Ingrid is the founder and CEO of bicycle retailer BIKEA. The company sells bicycles for customers to assemble themselves.

BIKEA has N different bicycles to offer. Ingrid wants to give each of them a human-readable name, to make it easy to remember. But doing this by hand is a very time consuming task.



You are given the number N , and your task is to generate N different names. To make the names readable, they must satisfy the following:

1. Each name has length between 3 and 20, and only consists of lowercase English letters.
2. Three consecutive letters of a name cannot all be vowels or consonants. Here we consider a, e, i, o, u vowels, while the remaining 21 letters are consonants.

For example, hello, abc, and lkab are all valid names, whereas ingrid, bo and louse are invalid.

Input

The input consists of one integer N ($1 \leq N \leq 30000$), the number of names to generate.

Output

Print N lines, each of them containing a name. It can be proven that it is possible to generate N different names.

Sample Input 1

3

Sample Output 1

abdullah
bjorn
nils

This page is intentionally left (almost) blank.

Problem B

Patrick's Triangle

Patrick is an unknown mathematician that wanted to become famous. Patrick had heard that the mathematician Blaise Pascal had become famous because of his triangle, Pascal's triangle.

$$\begin{array}{cccccc}
 & & & & & 1 \\
 & & & & 1 & 1 \\
 & & 1 & 2 & 1 & \\
 & 1 & 3 & 3 & 1 & \\
 1 & 4 & 6 & 4 & 1 &
 \end{array}$$

Patrick thought that if he could come up with his own triangle, Patrick's triangle, then he maybe too would become famous. After thinking about it for a long while, he came up with the following construction:

1. Along both the left side and right side of the triangle put the triangle numbers 1, 3, 6, 10, 15, 21, . . .

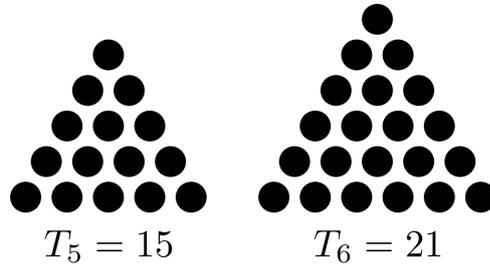
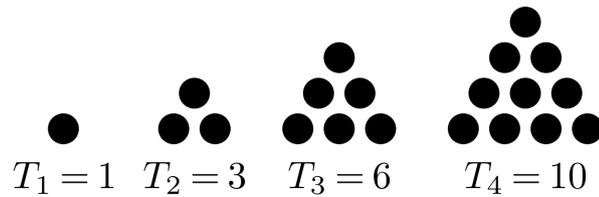
$$\begin{array}{cccccc}
 & & & & & 1 \\
 & & & & 3 & 3 \\
 & & 6 & & 6 & \\
 & 10 & & & 10 & \\
 15 & & & & & 15
 \end{array}$$

2. Fill in the inside of the triangle by adding the number above and to the left with the number above and to the right, in the same way as Pascal's triangle is constructed.

$$\begin{array}{cccccc}
 & & & & & 1 \\
 & & & & 3 & 3 \\
 & & 6 & & 6 & 6 \\
 & 10 & 12 & & 12 & 10 \\
 15 & 22 & 24 & & 22 & 15
 \end{array}$$

Patrick knew that in order for his triangle to make him famous, he needed to make sure no one else had invented it before him. So he searched through the On-Line Encyclopedia of Integer Sequences (OEIS.org), and luckily his triangle was missing from OEIS.

Now there was just one thing left to do, to upload his triangle to OEIS and become famous. To do this, Patrick calculated the first 10^6 rows of the triangle by hand. But before he uploaded his triangle to OEIS, he wanted to make sure there were no errors in his calculations. Can you help him check his calculations?



The first 6 triangle numbers. Wikimedia Commons, CC BY-SA 3.0.

Input

The first line consists of one integer Q ($1 \leq Q \leq 2 \cdot 10^5$), the number of queries. The next Q lines consist of three space separated integers N, K, X ($1 \leq K \leq N \leq 10^6$, $0 \leq X < 10^9 + 7$), where X is the value Patrick got when he calculated the K -th number on the N -th row in Patrick's triangle mod $10^9 + 7$. Patrick wants to know if X is correct or not.

Output

For each of the Q queries, output a single line with the string `Correct` if the K -th number on the N -th row in Patrick's triangle is equal to $X \bmod 10^9 + 7$, otherwise output the string `Incorrect`.

Sample Input 1

Sample Output 1

5	Correct
4 1 10	Incorrect
4 4 15	Correct
3 2 6	Correct
6 2 37	Correct
123456 61728 664742090	

Problem C

99 Problems

Ingrid is the founder of a company that sells bicycle parts. She used to set the prices of products quite arbitrarily, but now she has decided that it would be more profitable if the prices end in 99.

You are given a positive integer N , the price of a product. Your task is to find the nearest positive integer to N which ends in 99. If there are two such numbers that are equally close, find the bigger one.



Input

The input contains one integer N ($1 \leq N \leq 10^4$), the price of a product. It is guaranteed that the number N does not end in 99.

Output

Print one integer, the closest positive integer that ends in 99. In case of a tie, print the bigger one.

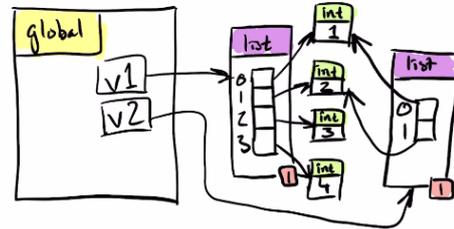
Sample Input 1	Sample Output 1
10	99
Sample Input 2	Sample Output 2
249	299
Sample Input 3	Sample Output 3
10000	9999

This page is intentionally left (almost) blank.

Problem D

Box and Arrow Diagram

What an embarrassment! Itaf got 0/5 points in her last “Fundamental programming in Python” exam. She studies Engineering physics at KTH and is struggling with this course. She is not alone, as 60% of her classmates failed the exam this year. The reason for this oddly high percentage is the so called *box and arrow diagram* (låd- och pildiagram).



An example of a box and arrow diagram, taken from github.com/dicander/box_arrow_diagram

In this part of the exam you are given a piece of Python code and you have to draw how the memory structure will look like when the program reaches a given line. Since Itaf is a high-rated competitive programmer her ego always came in the way whenever she tried to study for the test, because it felt “too easy”. But now she has become desperate and needs your help.

The *box and arrow diagram* is used to explain the memory structure inside Python. Simplified, the diagram can be seen as a directed graph with nodes (boxes) labeled from 1 to N and edges (arrows) labeled from 1 to M . The boxes corresponds to the objects in the memory of a Python program. Box 1 is special, it represents the *global* object. An arrow being drawn from box u to box v in the diagram means that object u stores a reference of object v . If u stores multiple references of v , then you draw multiple arrows from u to v . It is also possible for an object to contain references to itself.

An object u is said to be *alive* if there exists a path from the *global* object to u in the *box and arrow diagram*. Each object also has a reference counter. The reference counter of an object u is defined as the number of arrows (v, u) such that v is alive.

Itaf now needs your help, and she will ask you Q queries, each query can be one of two types.

- 1 X Remove the arrow with label X from the diagram.
- 2 Y Output the reference counter of the object with label Y .

Input

The first line consists of two space separated integers N, M ($1 \leq N, M \leq 2 \cdot 10^5$), where N is the number of boxes in the diagram and M is the number of arrows in the diagram.

The next M lines describe the arrows in the diagram. The i -th line contains 2 space separated integers U_i, V_i ($1 \leq U_i, V_i \leq N$), meaning the arrow with label i goes from box U_i to box V_i . Note that arrows forming loops and multi-edges are allowed.

The next line contains an integer Q ($1 \leq Q \leq 2 \cdot 10^5$), the number of queries. The next Q lines describe the Q queries. The j -th query is given as a pair of space separated integers C_j, X_j ($1 \leq C_j \leq 2$).

- If $C_j = 1$ then remove the arrow labeled X_j from the diagram ($1 \leq X_j \leq M$).
- If $C_j = 2$ then output the reference counter of object X_j ($1 \leq X_j \leq N$).

It is guaranteed that there will not be two queries of type 1 with same value of X_j , meaning the same arrow will never be deleted twice.

Output

For each query of type 2, output a single line containing the reference count of object Y_j .

Sample Input 1

```
3 4
1 2
2 3
1 2
3 3
7
2 2
2 3
1 4
2 3
1 1
1 3
2 3
```

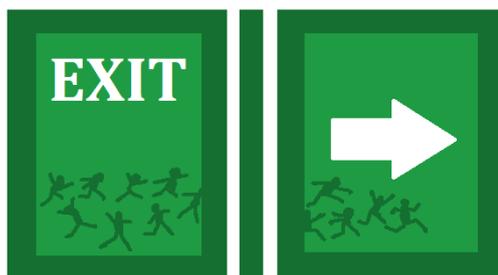
Sample Output 1

```
2
2
1
0
```

Problem E

Emergency Exit

Patrick wants to organize a big event in the computer rooms at KTH. He has set up posters and sent out invitations, and it looks like quite a large crowd of people will show up. Now Patrick has to deal with an unexpected problem: fire safety regulations. People must be able to get out fast enough in case of an emergency, so only a certain number of people are allowed in the computer rooms, and it looks like Patrick's event will exceed this number. But Patrick has not given up, he wants to prove that everyone can still get out fast enough if they behave optimally.



The building can be represented by an $N \times M$ grid where some cells are blocked and others are empty. There will be *exactly one* door to the building. This means that one cell on the edge of the grid is empty and the rest are blocked. Some empty cells have people in them. In one time step, a person can move to an empty cell that shares a side with the cell they are currently in. It is also possible to not move in a time step. Everyone moves simultaneously, and no two people may occupy the same cell at the end of a time step. When someone exits the grid (through the one door), they disappear and can no longer collide with other people.

Your task is to calculate the minimum time to get everybody out, and instruct the people to move in such a way to make this happen.

Input

The first line contains two integers N and M ($3 \leq N, M \leq 100$), the number of rows and columns in the grid.

The following N lines each contains a string of length M , representing each row of the grid. An empty cell without a person is denoted by “.”, a person is denoted by “P”, and a blocked cell is denoted by “#”.

Every cell on the edge of the grid will be blocked except for exactly one. The number of people in the grid will be between 1 and 100.

Output

If it is impossible for everyone to get out (i.e. someone is stuck inside the building), print one line with -1 .

Otherwise, print one line with the integer S , the number of seconds it takes to get everyone out. Let K be the number of people in the grid. Print K lines each containing one string of length S consisting of the characters “U” (up), “D” (down), “R” (right), “L” (left), “.” (wait). This string represents the actions the i -th person should take. The people are considered in the order they appear in the input, if the input grid is read row by row from left to right. When a person has exited the grid, give them wait instructions for the remaining time.

Sample Input 1

```

4 6
#####
#.P..#
#P.#P#
##P###

```

Sample Output 1

```

6
DDD...
.RDD..
ULLDDD
D.....

```

Sample Input 2

```

3 3
##.
#P#
###

```

Sample Output 2

```

-1

```

Sample Input 3

```

7 27
#####
#.....
#####.#####.#####.##
#.....#.....#.....#..#
#...#..#..P#...P#..P#..#..#
#...#..#P.P#P..P#...#.P#..#
#####

```

Sample Output 3

```

25
URUURRRRRRRRRRRRRRRR.....
LLLULUURRRRRRRRRRRRRR..
LLUUURRRRRRRRRR.....
.RRUURUURRRRRRRRRRRRRR.
UURUURRRRRRRRRRRRRRRR...
.UULUURRRRRRRRRRRRRRRR...
.LLLUULUURRRRRRRRRRRRRR
LUULLLLUURRRRRRRRRR.....

```

Problem F

Efficient Bus Routing

In the country of Optopia, they do everything as optimal as possible, and everything inside of Optopia has been designed with this in mind. For example, in order to guarantee that GPS route planners are always optimal, the road network of Optopia has been designed such that there is exactly one way to travel between any two cities in Optopia.

Optopia also has a network of bus lines, where each bus line travels back and forth between two cities, stopping at every city in between. Recently, there was a big scandal in Optopia. A citizen of Optopia figured out that the current network of bus lines is in fact non-optimal. It uses too many bus lines!

This is where you come into the story. You have been hired by Optopia to fix their embarrassing mistake. Your task is to design a new optimal network of bus lines. Optopia has specified two requirements for the new bus line network:

- It must be possible to travel between any two pairs of cities by bus.
- The number of bus lines should be as few as possible.

Model the country of Optopia as an undirected graph, where the nodes represent the cities of Optopia, and where the edges represent the roads between pairs of cities. Additionally, represent each bus line by two cities, the two end stops of that bus line.

Input

The first line consists of one integer N ($3 \leq N \leq 2 \cdot 10^5$), the number of cities in Optopia. The next $N - 1$ lines consists of two integers u and v ($1 \leq u, v \leq N$, $u \neq v$), a pair of cities that are connected by a road. It is guaranteed that Optopia is connected, meaning it is possible to travel between any two cities using the road network.

Output

On the first line output the optimal number of bus lines M . Then output M lines, where the i -th line should contain two space separated integers a_i, b_i ($1 \leq a_i, b_i \leq N$), the two endpoints of the i -th bus line.

Sample Input 1

```
3
1 2
2 3
```

Sample Output 1

```
1
1 3
```



SL bus line 4. Wikimedia Commons, CC BY-SA 4.0.

Sample Input 2**Sample Output 2**

5	2
1 2	2 4
1 3	3 5
1 4	
1 5	

Problem G

The Great Egg Hunt

Every year at Easter Bob's grandmother organizes *The Great Egg Hunt* in their family mansion. This has been a family tradition since even before Bob was born. Bob's grandmother starts by filling up a giant Easter Egg with candy. She then picks a room (uniformly) at random and hides the egg there. First person to find the egg gets to keep all the candy inside.



Easter Egg with Candy. Wikimedia Commons, CC BY-SA 3.0.

Bob's family mansion consists of N rooms. There are $N - 1$ doorways between pairs of rooms. You can assume that the mansion is connected, meaning it is always possible to walk from any room to any other room using the doorways.

As Bob is a veteran egg hunter, Bob has developed a method for finding the egg, which he calls *Egg First Search*:

1. If Bob is currently in an unexplored room, then he searches that room. Since Bob is a veteran egg hunter, if the egg is in that room, then he will find it.
2. Otherwise, Bob moves to an adjacent room that is in the direction of any of the closest unexplored rooms. If there are multiple rooms he could move to, then he picks one (uniformly) at random.

Assume it takes 1 unit of time for Bob to search a room, and also assume that it takes 1 unit of time for Bob to move from a room to an adjacent room.

The one thing Bob is still not sure about is in which room he should start his search. So he asks you for help. Given a map of the mansion, find which starting room(s) that minimize the expected time to find the egg using Bob's *Egg First Search* method.

Input

The first line consists of one integer N ($1 \leq N \leq 2 \cdot 10^5$), the number of rooms in the mansion. The next $N - 1$ lines consist of two space separated integers u and v ($1 \leq u, v \leq N, u \neq v$), a pair of rooms that are connected by a doorway. It is guaranteed that the mansion is connected, meaning it is always possible to walk from any room to any other room using the doorways.

Output

On the first line print an integer M , the number of optimal starting rooms. On the second line, print M space separated integers S_1, \dots, S_M ($1 \leq S_1 < S_2 < \dots < S_M \leq N$), the list of optimal starting rooms.

Sample Input 1

```
3
1 2
2 3
```

Sample Output 1

```
2
1 3
```

Sample Input 2

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

Sample Output 2

```
4
2 3 4 5
```

Problem H

Bar Classification

You are taking a course on machine learning at your university, and as homework you have been tasked with writing a program that can tell vertical bars from horizontal bars in images. To generate some training data, you use the following method. First, take an $N \times N$ grid, and fill it with zeros. Next, take a row or a column, and fill it with ones. Finally, take at most N arbitrary cells, and flip them. Flipping a cell means changing a zero to a one, or changing a one to a zero.



Generating data this way is easy, but how to generate all the answers? It will take hours to go through the training data manually. If only you had a program that finds all the outputs automatically somehow.

You are given an $N \times N$ matrix that has been generated as in the description. Write a program that finds whether it was a column or a row that was filled with ones, or if it is impossible to determine.

Input

The first line of input consists of an integer N ($2 \leq N \leq 1000$), the size of the grid.

The following N lines each contain a string of length N consisting of zeros and ones. These are the rows of the grid.

It is guaranteed that the input was generated by taking a grid of zeros, putting ones on a row or a column, and then flipping at most N cells.

Output

If the bar was vertical (a column), print “|”. If it was horizontal (a row), print “-”. If it is impossible to determine (because it could be both), print “+”.

Sample Input 1

```
5
01100
01000
01001
00000
01000
```

Sample Output 1

```
|
```

Sample Input 2

```
3
111
000
111
```

Sample Output 2

```
-
```

Sample Input 3**Sample Output 3**

3 010 101 010	+
------------------------	---

Problem I

Sperhling

On her spare time Caitlin loves to do speed typing. Unfortunately Caitlyn was never good at spelling. So what often happens is that Catelin makes a mistake spelling a word, and has to go back and fix her typo. One word in particular that Caitlynn has a hard time spelling is the word `mischievous`. Catelyn usually spells it `mischevious` instead. In order to fix this typo, Caytlin needs to do 12 key presses.

```

      mischevious|
←    mischeviou|s
←    mischevio|us
←    mischevi|ous
BS   mischev|ous
←    mische|vous
←    misch|evous
i    mischi|evous
→    mischie|vous
→    mischiev|ous
→    mischievo|us
→    mischievous|
→    mischievous|
```

Katelin thinks that no matter how much time she puts into speed typing, she will always make mistakes. But maybe she can get better at quickly fixing her mistakes. Katelyn would like you to help her with this.

Suppose that Caytlyn has just written the word S_1 , but that she should have written S_2 . Assume that the cursor is initially at the end of S_1 , and that after fixing the typo, Caitlin needs to place the cursor at the end of S_2 . Output the fewest number of key presses needed to make S_1 into S_2 .

The keys you are allowed to use are

- the left and right arrow keys.
- the back space key.
- the keys `a, . . . , z`.

You are not allowed to press back space or the left arrow key when the cursor is all the way to the left. You are also not allowed to press the right arrow key when the cursor is all the way to the right.

Input

The first two lines contain the strings S_1 respectively S_2 . Both S_1 and S_2 consist only of lowercase English letters, and are between 1 and 100 characters long.

Output

Output a single line with the fewest number of key presses needed to make S_1 into S_2 .

Sample Input 1**Sample Output 1**

mischevious mischievous	12
----------------------------	----

Sample Input 2**Sample Output 2**

cerstermergerd customer	20
----------------------------	----

Sample Input 3**Sample Output 3**

caitlin caitlynn	5
---------------------	---