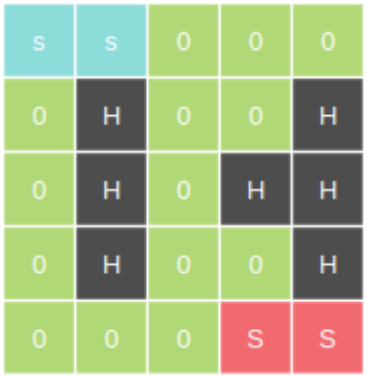
SofaProblem

Problem Description

Leo is trying to move his sofa to different place in his house. The issue is that the house is filled with lot of household items already.

His home will be divided into cells of M rows and N columns like a matrix. The matrix is used to represent the house and the things in it. The current placement of sofa will be represented as "s" and place where the sofa is required will be represented as "S". "H" represents the presence of household items while "0" indicates empty spaces where the Sofa can be moved. The sofa will always occupy two cells and can either move vertically or horizontally. The sofa can be rotated, but the rotation can only occur within a 2*2 area of free cells that includes the sofa. The destination to where the sofa needs to be moved will always be free of household items.



Since sofa is heavy object, it must only be moved and not lifted. Help Leo to find the minimum possible steps required to move the sofa to the required place without altering the positions of other household items. Moving the sofa one cell as well as rotating the sofa are considered as one step.

Constraints

4 < M, N < 25

Input

First line contains two space separated integers representing M, N.

Next M lines contain N space separated characters being either "s", "S", "H" or "0" representing the house's matrix.

Output

Print a single integer representing minimum number of steps required to move the sofa to the required place. Print "Impossible" if it is impossible to move the sofa to the required place.

Time Limit (secs)

1

Examples

Example 1

Input

5 5
s s 0 0 0
0 H 0 0 H
0 H 0 H H
0 H 0 0 H
0 0 0 S S

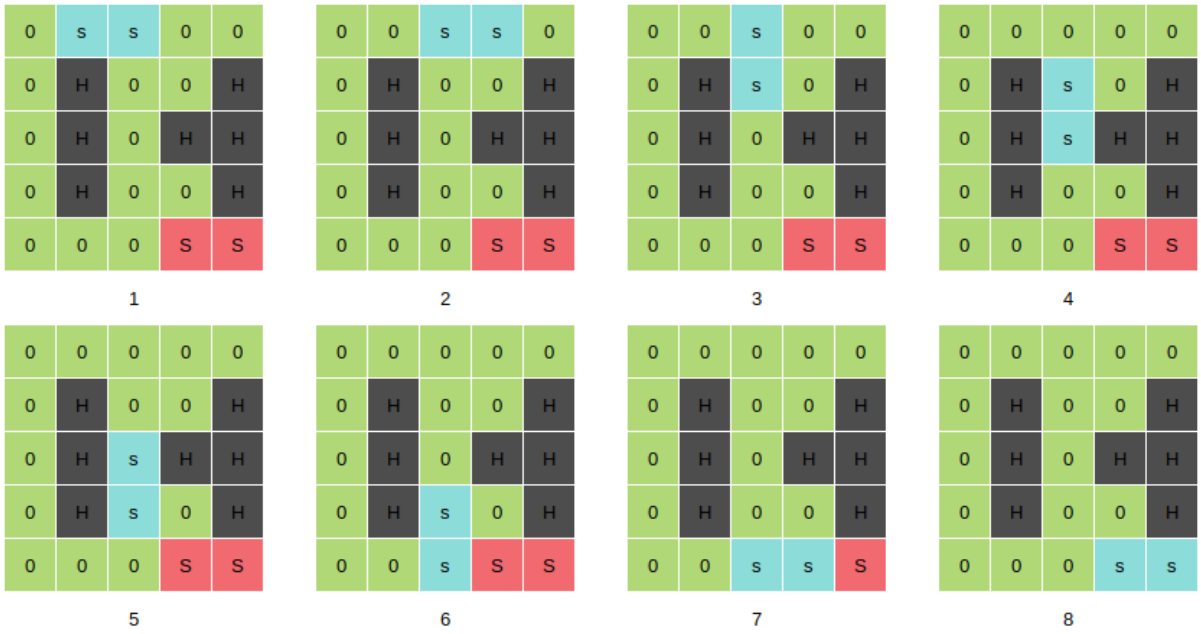
Output

8

Explanation

The diagram shown in the *description* depicts the input given

Below diagrams show the movement of sofa to the required place.



After two moves, the sofa occupies a 2x2 area of free cells, allowing it to be rotated. Following this process, the sofa has been moved to its destination in 8 steps, which represents the minimum required.

Example 2

Input

5 5
s s 0 H 0
0 H 0 0 H
0 H 0 H H
0 H 0 0 H
0 0 0 S S

Output

Impossible

Explanation

As you can see it impossible to move the sofa to the required place without rotating and there is not enough place to rotate the sofa, hence print "Impossible".

Formatting Software

Problem Description

Vivek is developing software that takes a set of numerical instructions, processes them, and formats them as required.

Vivek is working on a big project, and he asked you to implement a software that takes a set of numerical information in the form of r x c matrices, processes them, and formats them according to the given instructions. The instructions consist of multiple lines, with each line containing one or more integer values. Each integer specifies which matrix should be printed.

For example, if the instructions are like below,

4

1 2 3

which says that you should first print matrix number 4 from the information provided. Then, on the subsequent lines, you need to print matrices 1, 2, and 3 sequentially.

Given N, representing the number of matrices, and two additional integers r and c representing the number of rows and columns in each matrix, along with the N matrices provided contiguously over r lines, and in the decribed format, print the formatted information.

(Refer *Examples* section for better understanding of the input)

Constraints

1 <= N <= 10

1 <= r, c <= 10

1 <= numbers in the matrix <= 100

1 <= number of matrices in the instructions <= 20

Matrices are numbered from 1 to N; thus, the instructions will include only numbers within this range. Note that numbers may be repeated.

Input

First line consists of an integer N, denoting the number of matrices

Second line consists of two space separated integers denoting the number of rows and columns in each matrix.

The next r lines contain the N matrices, each of size r × c, listed sequentially. Each matrix will have integers separated by spaces, and consecutive matrices will be separated by a space as well.

Remaining lines represent the instructions of how the information should be formatted.

Output

Print the formatted information following the given instructions.

Time Limit (secs)

1

Examples

Example 1

Input

7

1 3

5 4 12 11 19 3 0 15 7 1 2 3 4 5 6 10 9 8 3 60 71

5

7 1 3

2 4 5

3 6

4

Output

4 5 6

3 60 71 5 4 12 0 15 7

11 19 3 1 2 3 4 5 6

0 15 7 10 9 8

1 2 3

Explanation

If you split the given data into 1x3 submatrices and print them in the specified format, you will obtain the output. Let's go over the input processing.

Line 4 i.e. input 5 asks to print matrix number 5. Output 4 5 6 corresponds to this input.

Line 5 i.e. input 7 1 3 asks to print matrices 7, 1 and 3. Output 3 60 71 5 4 12 0 15 7 corresponds to this input.

Similarly processing line number 6, 7 and 8 of the input, we obtain lines 3, 4 and 5 of the output.

Example 2

Input

4

2 3

1 2 3 4 5 6 7 8 9 10 11 12

12 11 10 9 8 7 6 5 4 3 2 1

1 2 3

4 1

2 3

3

1 4

Output

1 2 3 4 5 6 7 8 9

12 11 10 9 8 7 6 5 4

10 11 12 1 2 3

3 2 1 12 11 10

4 5 6 7 8 9

9 8 7 6 5 4

7 8 9

6 5 4

1 2 3 10 11 12

12 11 10 3 2 1

Explanation

If you split the given data into 2x3 submatrices and print them in the specified format, you will obtain the output. Let's go over the input processing.

Line 5 i.e. input 1 2 3 asks to print matrices 1, 2 and 3. Output lines 1 and 2 corresponds to this input.

Line 6 i.e. input 4 1 asks to print matrices 4 and 1. Output lines 3 and 4 corresponds to this input.

Similarly processing line number 7, 8 and 9 of the input, we obtain lines the remaining lines of the output.

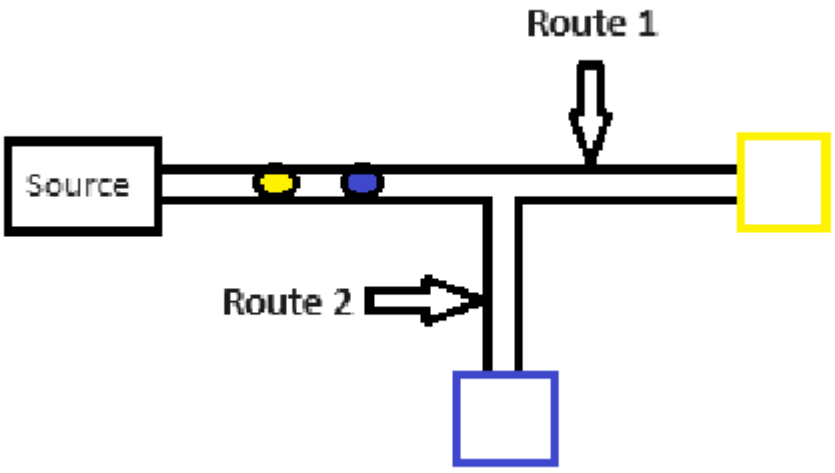
Route The Balls

Problem Description

Feeling bored at home, Lulu decided to play a mobile game called "Route The Balls". In the game, a source continuously releases different coloured balls, and the objective is to sort them into their corresponding-coloured buckets.

Between the source and the buckets, there are several junctions. The source, buckets are also considered as junctions. Initially, all the paths will be closed. At each junction, when multiple paths are present, only one path can be opened at a time. From a given junction, if a path is opened, any other paths that is currently open from the junction will be automatically closed.

For instance, if there exists a T-shaped junction, refer to the diagram below.



The initial ball is blue, and it needs to be directed towards the blue bucket positioned below. Initially, all sides of the paths are closed. Therefore, you open the downwards path (source - blue bucket) to allow the ball to flow into blue bucket. The subsequent ball is yellow, requiring it to be directed towards the yellow bucket situated to the right. To achieve this, you open the path on the right side (source - yellow bucket), automatically closing the previous downward opening. Hence, the junction was opened twice in total.

Given the network of junctions, their connections to each other, and the sequence of the balls, determine the total number of junction openings required to guide the balls into their respective buckets.

Constraints

1 <= number of balls <= 50

1 <= number of junctions <= 50

1 <= length of name of colours and junctions <= 20

There will always be a single source.

There will be only one bucket for each colour.

No balls will be used to route from the source if the corresponding-coloured bucket is not available.

Name of the junctions will consist of lower-case alphabets and digits.

Input

First line consists of *N,* denoting the number of lines representing the routes structure of the game.

In the following N lines, each line contains space-separated elements. The first element denotes a junction, while the subsequent elements indicate its connections to other junctions. Note that the paths are unidirected from source to buckets.

The last line comprises the colours of the balls originating from the source, separated by spaces.

Output

Print the total number of paths you need to open to route all the balls into respective buckets.

Time Limit (secs)

1

Examples

Example 1

Input

6

source jun1 jun3

jun1 jun2

jun2 jun4 jun6

jun4 blue

jun6 pink yellow

jun3 red

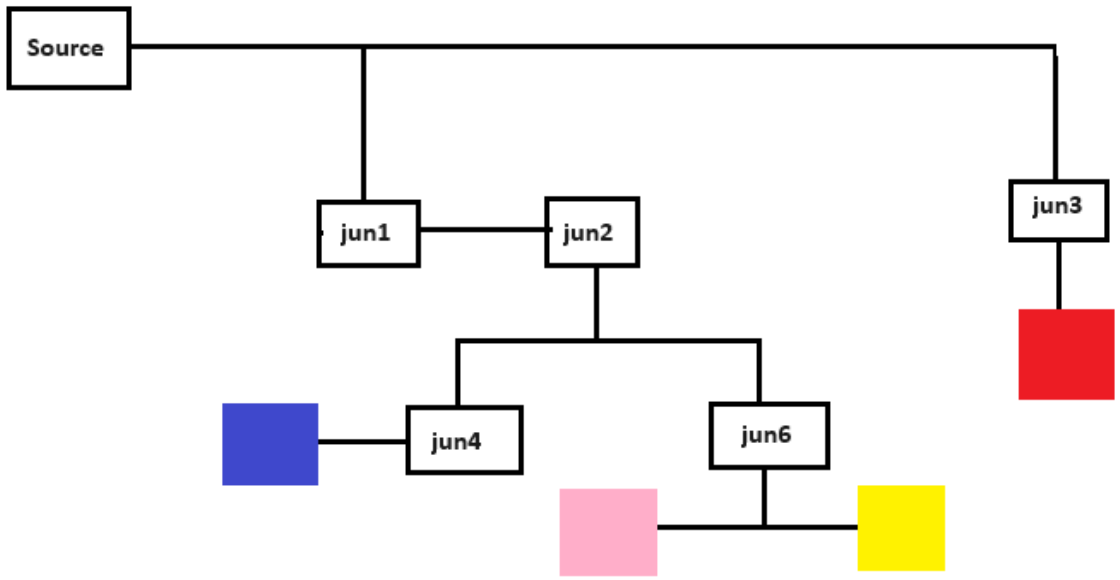
blue yellow red pink blue

Output

11

Explanation

The given input, when visualised looks below.



Initially, every path is closed.

First ball is blue coloured. For this, to land in blue bucket, open the paths, source - jun1, jun1 - jun2, jun2 - jun4, jun4 - blue. Total number of paths opened = 4

Next ball is yellow coloured. The paths from source - jun2 is already open. Open the paths jun2 - jun6 and jun6 - yellow. Total number of paths opened = 4 + 2 = 6

Next ball is red coloured. For this, open the paths source - jun3, jun3 - red. Note that the path which was open earlier from source (source - jun1) will be closed. Total number of paths opened = 6 + 2 = 8

Next ball is pink coloured. For this, open the paths source - jun1 and jun6 - pink. Note that the path which was open earlier from source (source - jun3) will be closed. The paths from jun1 - jun2, jun2 - jun6 are already open. Total number of paths opened = 8 + 2 = 10

The last ball is blue coloured. The paths source - jun1, jun1 - jun2, jun4 - blue is already open. Open the path jun2 - jun4. Total number of paths opened = 10 + 1 = 11

Hence, print 11.

Example 2

Input

4

source jun1

jun1 violet jun2 jun3

jun2 red

jun3 green yellow

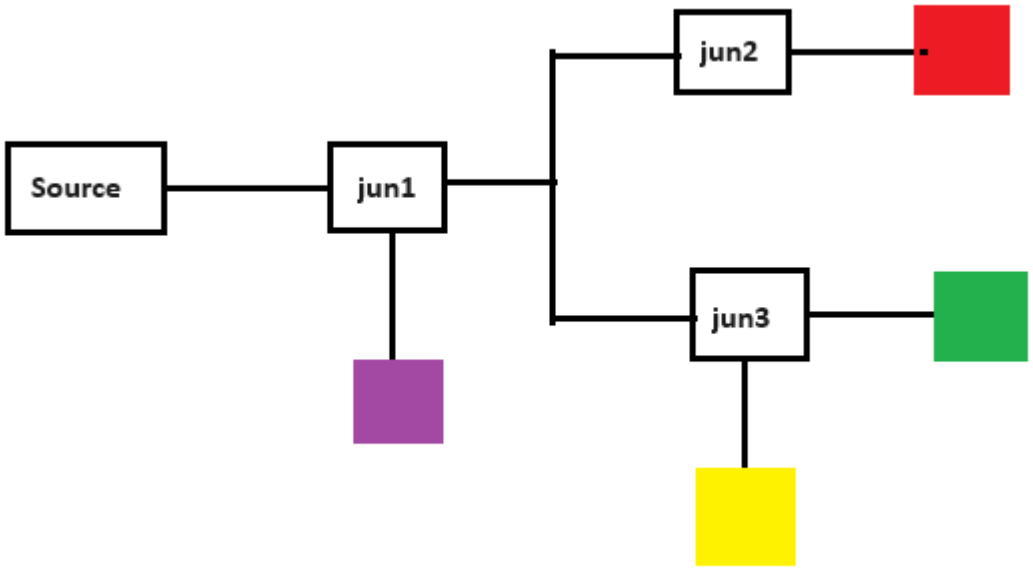
green yellow red green green violet

Output

9

Explanation

The given input, when visualised looks below.



Initially, every path is closed.

The first ball is green coloured. Open the paths, source - jun1, jun1 - jun3, jun3 - green. Total number of paths opened = 3

Next ball is yellow coloured. The paths source - jun1, jun1 - jun3 are already open. Hence open the path jun3 - yellow. jun3 - green will be closed. Total number of paths opened = 3 + 1 = 4

Next ball is red coloured. Open the paths jun1 - jun2 and jun2 - red. Other paths which are opened from the same junctions will be closed. Total number of paths opened = 4 + 2 = 6

Next ball is green coloured. Open the paths jun1 - jun3 and jun3 - green. Total number of paths opened = 6 + 2 = 8

Next ball is also green, and the respective paths are opened in the above steps. Hence the total paths opened remains same as of above step.

Next ball is violet. Open the path jun1 - violet. Total number of paths opened = 8 + 1 = 9

Hence, print 9.

ReenuCircuit

Problem Description

Reenu is an electrical engineer who frequently designs new circuits. She constructs circuits using two types of resistors: horizontal and vertical, along with a junction. In each circuit, there are two positions: an opening and a closing. The circuit is connected to the power supply at the opening position, and the current flows through the resistors and junctions until it reaches the closing position.

In the matrix representation of the circuit, power can only flow vertically (up or down) if a vertical resistor is present, and it can only flow horizontally (left or right) if a horizontal resistor is present. The junction connects all four sides. The vertical and horizontal resistors have 1 units of resistance, while the junction considered to be having low resistance, **ignorable**.

Vertical resistor is represented by "|" (pipe), horizontal represented by "-"(hyphen) .a "." (period) symbol represents terminals and "+" (plus) representing junction.

The resistance for series connection is RT = R1 + R2 and for parallel connection is 1/RT = 1/R1 + 1/R2.To know about resistance in series and parallel look here.

Given the circuit, reduce and determine the equal resistance of the circuit between opening position and closing position.

To know more about reduction of resistor in series and parallel look here

Constraints

3 <= N <= 10

Input

First line consists of *N,* denoting the number of rows, columns in the matrix.

The following N lines represent the circuit as a matrix.

Output

Print the total resistance of the circuit.

Time Limit (secs)

1

Examples

Example 1

Input

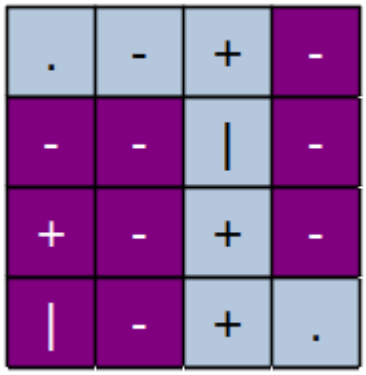
```
4
.-+-
--|-
+-+-
|-+.
```

Output

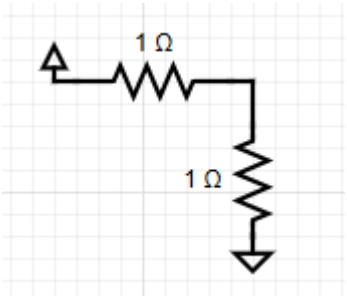
2

Explanation

The above input is visualized below.



The Equivalent circuit of this will be,



Two resistors in series and adding them results in 2 units of resistors which is the equivalent.

Example 2

Input

5
.-+-+
--+-.

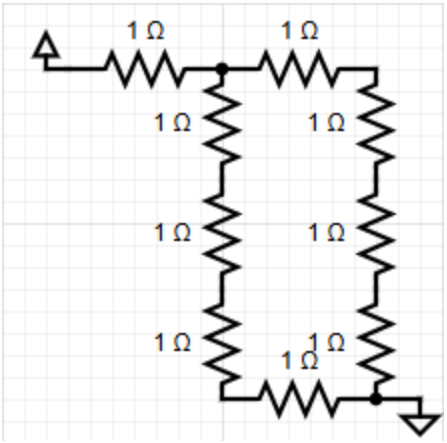
Output

3

Explanation

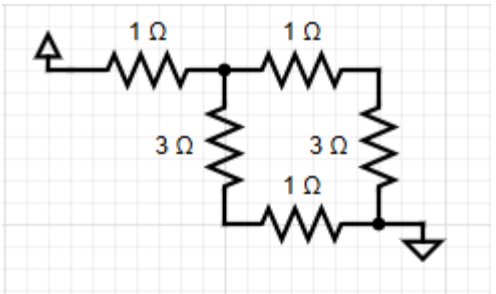
.	-	+	-	+
-	-	|	-	|
-	-	|	-	|
-	-	|	-	|
-	-	+	-	.

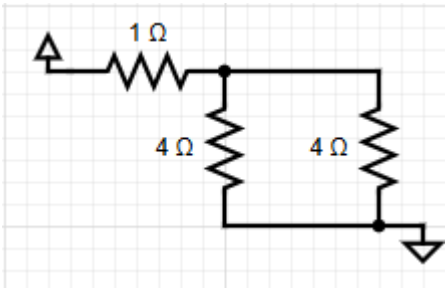
The equivalent circuits look as below.

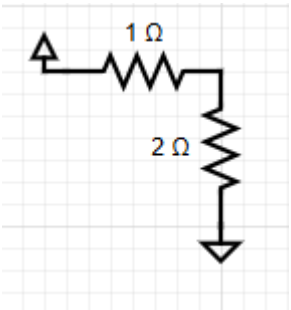


The reduction goes as below...

First the three resistors series on both the branches will combined together.Followed by 3 and 1 unit resistors on both the branches will combined together.Forming 4 unit of resitors in parallel.The 4 units of resistors are combined and become a single 2 units of resistors







The resultant 2 units of resistor is series with 1 unit of resistor equivalent to 3 units of resistors.And thus the equivalent resistor 3.