Problem A. Rain Diary

Time limit: 1 second
Memory limit: 512 megabytes

Petya lives in Saint Petersburg and he is keen on meteorology. It is widely believed that it constantly rains in Saint Petersburg. Petya has decided to statistically prove or disprove this statement. For this Petya started a rain diary and every week at Sunday he makes a record, whether it is rainy or not. Petya signs every record with a number of current day in a month, days in a month are numerated from 1 to number of days in the current month (from 28 to 31 in a different monthes).

Today Petya opened his diary and realized, that he made the last record two weeks ago and forgot to make a record last Sunday. Petya remembers, that there was a heavy rain a week ago. He decided to make two records: for today and for the last Sunday. He knows the number of the current day n and sees the number m in the two weeks ago record. What is the number Petya should sign his record for the last Sunday?

Input

The first line of the input contains two integers n and m ($1 \le n, m \le 31$) — the number of the current day of the month and the number that the record was signed two weeks ago.

Output

Output a single number – what number should the record be signed for the last Sunday.

Example

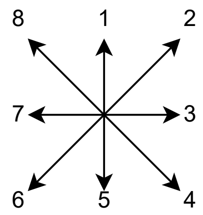
standard input	standard output
9 24	2

Problem B. Magnetic Games

Time limit:	1 second
Memory limit:	512 megabytes

Vova has a magnet which is his favorite toy. Vova likes to play with it on a big field which can be represented as rectangle of n cells height and m cells width.

One day he came to the field with his magnet and lost it in one of the cells. Vova was frustrated, but found a solution — he puts a compass in each cell of the field. Compass arrow may be oriented in one of the following 8 directions. Each direction has its own number:



Arrows take direction close to the direction to the magnet: if the magnet is strictly along the diagonal to the compass, arrow of this compass points diagonally, otherwise — it points in the closest of the vertical and the horizontal direction to the magnet. Arrow of the compass in the cell with the magnet may be directed anywhere.

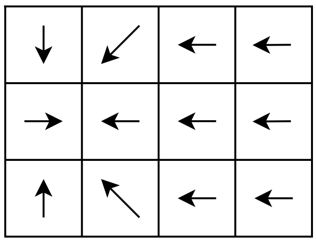


Illustration for the sample test, arrows of all compasses point to the magnet.

Vova was ready to start searching, but suddenly recalled that there is an anomaly at the field, that inverts compasses along one horizontal and one vertical line. Arrows of the comapasses in the anomaly zone point in the opposite direction to the correct one.

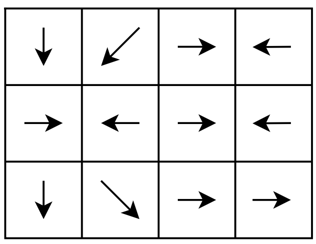


Illustration for the sample, readings of some of the compasses are inverted.

Now Vova doesn't know how to find his magnet and asks you for help.

Input

The first line of the input contains two integers n and m — size of the field ($2 \le n, m \le 1500, nm \ge 6$).

The following n lines contain information about compasses on the field.

The i-th line contains m integers $a_{i,j}$ ($1 \le a_{i,j} \le 8$), the j-th number on the i-th line describes the direction of the compass arrow in the j-th cell on the i-th line.

Output

The first line of output should contain two integers x and y — the row and the column of the cell where the magnet is.

The second line must contain two integers a and b — which the horizontal and which vertical line are affected by the anomaly.

Example

standard input	standard output
3 4 5 6 3 7 3 7 3 7 5 4 3 3	2 1 3 3

Note

The magnet in sample input is in the cell $(2, 1)$, the anomaly inverts horizontal number 3 and vertical number 3.

Problem C. Campfire Riddle

Time limit: 1 second
Memory limit: 512 megabytes

One day Yurik found himself in a forest near the campfire where n people had gathered.

It turned out that some of them are friends. Let's number all people with integers from 1 to n. Let's denote the number of people that are friends with the i-th person, as d_i. It suddenly turned out that two people with numbers i and j ($i \neq j$) are friends if and only if $d_i = d_j$.

When Yurik came home he got curious, what is the minimum number of pairs of people that could be friends, so that the condition described above to be satisfied?

Input

The only line contains one integer n ($1 \leq n \leq 5\,000$) — the number of people.

Output

Print one integer — the minimum number of pairs of friends.

Examples

standard input	standard output
4	3
5	4

Note

Consider the first example. All possible cases are described below:

1. Each two people are friends with each other. In this case the number of pairs of friends is $\frac{4 \cdot 3}{2} = 6$.
2. Three people are pairwise friends with each other, and the fourth person is not friend with anyone. In this case the number of pairs of friends is 3.

Problem D. The Tree

Time limit: 1 second
Memory limit: 512 megabytes

You are given an infinite binary tree. The tree has the root and infinitely many vertices, each vertex has left son and right son, each vertex except the root has a parent.

Every vertex can be painted in one of c colors or be colorless. Initially, all vertices are colorless.

You need to process two types of requests:

1. `color(`u`, `x`)` You are given a vertex u, paint vertex u with color x, and then call `color(`L`, `$(x+1) \bmod c$`)` for its left son L and `color(`R`, `$(x-1+c) \bmod c$`)` for its right son R. Note that this operation repaints the entire (infinite) set of vertices in the subtree of the vertex u. Here mod is the operation of taking a division remainder. If the vertex has already been painted, then its color changes to the new one.

2. Given a vertex, output its current color.

Input

The first line contains two integers q, c — the number of queries and colors, respectively ($1 \le q \le 5 \cdot 10^5$, $1 \le c \le 10^9$). This is followed by q queries, each starts with an integer t_i — the type of the i-th query.

If $t_i = 1$, then an integer x ($0 \le x \le c-1$) is given in the line, the color that the vertex of the query u should be colored. The next line describes the path to the vertex u in the form of a non-empty string s_i consisting of the characters '`L`' and '`R`'. This string specifies the path from the root of the tree to the vertex u, where '`L`' denotes going to the left son, and '`R`' — going to the right son.

If $t_i = 2$, then the next line specifies the path to the vertex whose color should be output, described similarly to the previous query.

It is guaranteed that the sum of the lengths of the paths to all the vertices of the queries does not exceed $5 \cdot 10^5$.

Output

For each request of the second type print the color of the corresponding vertex on a separate line. If the vertex is colorless, print -1.

Example

standard input	standard output
6 3 1 2 L 2 LL 1 0 LL 2 LLR 2 LR 2 R	0 2 1 -1

Problem E. Just Like Pickle

Time limit: 1 second
Memory limit: 256 megabytes

Grasshopper is standing on a line at a point with coordinate 0. In one turn it can choose any non-negative integer number k and jump to the left or to the right to the distance 2^k.

Help it to find out what is the minimum number of turns it has to do in order to move from the point with coordinate 0 to the point with coordinate x.

Input

The first line contains a single integer t — a number of test cases ($1 \le t \le 100\,000$).

Each test case consists of a single line, that contains a single integer x — coordinate of a target point for the grasshopper ($-10^{18} \le x \le 10^{18}$).

Output

For each test case output a single integer — the minimum number of turns that the grasshopper has to do.

Example

standard input	standard output
5 1 -4 0 -7 239	1 1 0 2 3

Problem F. Lazy to Win

Time limit: 1 second
Memory limit: 512 megabytes

Alexey is probably the smartest and at the same time the laziest person in the world. Today he is participating in the Olympiad.

At the Olympiad, participants are given n problems. The participant will receive a_i points for the correct solution of the i-th problem. No points are given for an incorrect solution. The winner's diplomas will be awarded to those participants who receive at least half of the total number of points. For example, if there are given three problems at the Olympiad, the cost of which in points are 1, 3 and 4, it is enough to receive four points to be awarded the winner's diploma.

Alexey came to the Olympiad to get a winner's diploma. Alexey is very smart and can correctly solve any problem at the Olympiad. But Alexey is also very lazy and wants to solve as few problems as possible.

Alexey is so lazy that he even lazily chooses tasks that he will solve. He wants to choose some problem with the number k, and then solve problems with the numbers $k, k+1, k+2\ldots$ until he has enough points to get a winner's diploma. The maximum that Alexey is ready for is to skip one problem and not solve it, in order to solve even fewer problems in the end.

Determine the minimum number of problems that Alexey needs to solve using his strategy to get the winner's diploma.

Input

On the first line there is one integer n — the number of problems that were given at the Olympiad ($1 \le n \le 10^5$).

On the second line there are n integers $a_1, a_2, \ldots a_n$ — the cost of each problem in points ($1 \le a_i \le 10^9$).

Output

On the first line output single integer — the minimum number of problems that Alexey needs to solve using his strategy to get the winner's diploma.

Examples

standard input	standard output
5 1 1 2 1 2	2
3 2 3 1	1

Note

In the first example, participants need receive at least four points to get the winner's diploma. Alexey can start solving the third problem, then skip the fourth problem and get four points by solving two tasks.

In the second example, it is enough to solve only the second problem and receive three points.

Problem G. The Length of the Sequence

Time limit: 1 second
Memory limit: 512 megabytes

Consider the segment of non-negative integers from l to r. Write them in a row in decimal notation, getting a string a. For example, if $l = 3$ and $r = 10$, $a = 345678910$.

You have to find such segment of consecutive non-negative integers $[l, r]$ ($0 \le l \le r \le 10^{18}$) that the length of the string a, corresponding to this segment, is exactly S, and the number of integers in the segment $[l, r]$ is maximum possible.

Input

The only line contains one integer S ($1 \le S \le 10^{18}$).

Output

Print the length of the optimal segment $[l, r]$ in the first line. If there is no solution, print -1.

If the solution exists, print two integers l and r in the second line.

If there are multiple optimal solutions, print any of them.

Examples

standard input	standard output
3	3 0 2
10	10 0 9
20	15 0 14

Problem H. Pines

Time limit: 1 second
Memory limit: 512 megabytes

In order to prepare for the celebration in the city P, it's been decided to decorate the alley. Two teams were hired for this, one is responsible for illuminating the alley, the other is responsible for planting the alley with pines.

Alley can be represented as a line. They decided to decorate it as follows — starting with a pine tree, alternate between trees and lamps. As a result, $n + 1$ pines will be planted on the alley and n lamps will be installed.

Lamps were installed almost instantly, and they were of two types — "A" and "B". "B"-type lamps always shine white light, color of "A"-type lamps on the other hand depends on its surroundings. If a tree standing to the left of the lamp is higher than a tree standing to the right of the lamp, then the lamp lights up red, otherwise it lights up blue.

When the pines have finally arrived it has turned out that all their heights are distinct and take values from 1 to $n + 1$. It's been decided to arrange pines so that the number of red and the number of blue lamps were as close to each other as possible.

Help the team responsible for planting with arranging all $n + 1$ pines so that the difference between the number of red lamps and the number of blue outs was minimal possible. Formally, if after the planting there are r red and b blue lamps, it's required to minimize the value of $|r - b|$.

Input

The first line of input contains one integer n — the total number of lamps ($1 \le n \le 2 \cdot 10^5$).

The second line of input contains n characters, i-th equal to "A" or "B" - the type of i-th lamp.

Output

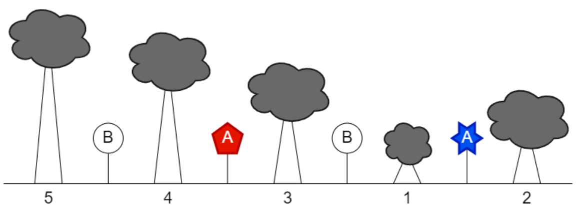
Print $n + 1$ unique numbers from 1 to $n + 1$ — heights of pines in the optimal placement. If there are several optimal answers, you can print any of them.

Examples

standard input	standard output
2 AA	1 3 2
4 BABA	5 4 3 1 2

Note

Illustration for the second sample test:



For clarity, red lamps in the illustration have a pentagon shape and blue lamps have a star shape.

Then $r = 1$, $b = 1$, $|r - b| = 0$ and this arrangement will be one of the most optimal.

Problem I. Circus Performance

Time limit: 1 second
Memory limit: 512 megabytes

A very famous circus is in the city! There are exactly n acrobats in the circus, and this time they prepared a special show in honor of the SPb Team School Programming Olympiad 2022.

It is known that the i-th acrobat has a height equal to a_i and weight equal to b_i. Any three acrobats can get together and perform an unusual stunt. If acrobats with numbers i, j and k perform a stunt, the *efficiency* of the stunt is estimated as $a_i b_j + a_j b_k + a_k b_i$.

A circus' trainer considers an ordered trio of acrobats (i, j, k) *good* if the efficiency of their stunt is no less than if they are arranged in reverse order (k, j, i).

For the final act of the show the trainer wants line up all n acrobats so that any triple of consecutive acrobats would be good. Help him with this difficult task!

Input

The first line of input contains an integer n representing the number of acrobats in the circus ($3 \le n \le 1000$).

In the i-th of the following n lines there are two space-separated integers a_i and b_i — the height and weight of the i-th acrobat ($1 \le a_i, b_i \le 10^9$).

Output

Print n different integers from 1 to n — the numbers of acrobats in the order in which they should be lined up.

Examples

standard input	standard output
3 10 70 30 40 50 60	2 3 1
4 99 99 11 11 88 88 55 55	2 4 3 1

Problem J. Square Running

Time limit: 1 second
Memory limit: 512 megabytes

Runsberg citizens are keen on sports, especially running. Standard marathons bored them, so they decided to organize a marathon in Minecraft style. The event will be held on Runsberg's Central Square. Citizens believe that the most important in sports is not to win, but to participate, so the goal of the square running is not to win, but to take a beautiful photo of all participants.

The Runsberg's Central Square is a rectangular square divided into unit squares. Rows are enumerated from top to bottom starting from one, columns are enumerated from left to right starting from one. Each unit square has coordinates r and c — row and column number, respectively.

There is a rectangular grass field on the Square. The sides of the grass field are parallel to the sides of the Square. The left top unit square of the grass field has coordinates (R_L, C_L), and the right bottom unit square of the grass field has coordinates (R_R, C_R). There are n lanes for n runners around the grass field. Lane i is at the distance i from the grass field's border, there is a runner with the number i on lane i. Runner i starts from the unit square (r_i, c_i). All runners start at the same time at the same speed: every second each athlete moves from the current square in his lane to the next square in his lane in a counterclockwise direction.

There is a photographer on the rectangular grass field. The photographer stays on the unit square with coordinates (R_p, C_p). The photographer wants to take a beautiful photo. He tests an innovative dual-lens camera. This camera takes a photo in two opposite directions simultaneously. The photographer considers a photo beautiful if all the runners at the time he makes a photo are in the same row R_p or in the same column C_p. Thanks to the innovative camera, the runners can be in the same row with the photographer both to the right and to the left of him, or in the same column with the photographer both above and below him.

Your task is to find out what is the minimum number of seconds t after the start of the marathon that the photographer will be able to take a beautiful photo or to find out that it is impossible to take a beautiful photo under these conditions.

Input

The first line contains integer n ($1 \le n \le 18$) — number of the runners. In the second line there are six integers R_L, C_L, R_R, C_R ($n+1 \le R_L \le R_R \le 100-n$, $n+1 \le C_L \le C_R \le 100-n$), R_p ($R_L \le R_p \le R_R$), C_p ($C_L \le C_p \le C_R$) — the left top unit square coordinates, the right bottom unit square coordinates, the photographer's coordinates, respectively. It is guaranteed, that $R_R - R_L + C_R - C_L$ is divisible by 4.

In the next n lines there are two integers r_i, c_i — start coordinates of the runner i. It is guaranteed, that the start coordinates of the runner i are on the lane i, there is only one runner on each lane and the lane i is in the distance i from the grass field's border.

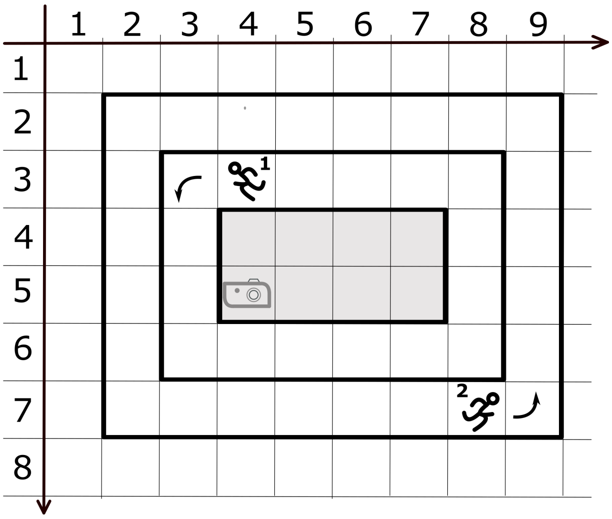
Output

Output the minimum number of seconds t after the start of the marathon that the photographer will be able to take a beautiful photo or -1, if it is impossible to take a beautiful photo under this conditions.

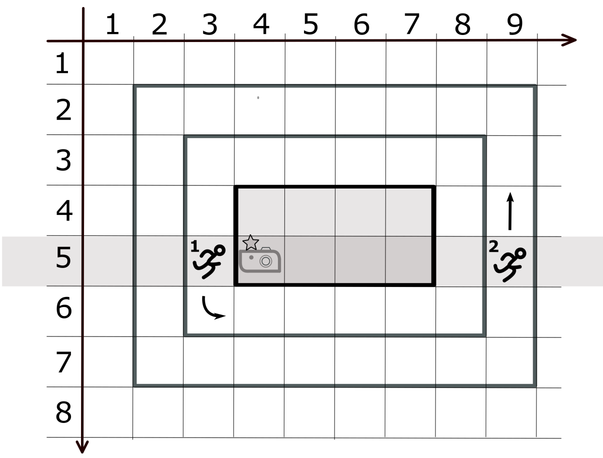
Examples

standard input	standard output
2 3 3 5 5 4 3 2 3 1 1	3
2 4 4 5 7 5 4 3 4 7 8	3

Note



Staring positions of the runners.



Positions of the runners after 3 seconds. All runners are in the R_p.
So the photographer is able to take a beautiful photo.

Problem K. Pick a Pair

Time limit: 1 second
Memory limit: 512 megabytes

Friends play an interesting game with words. The goal of this game is arranging words in pairs.

Friends have n words of the same length. They are choosing the largest number k such that it is possible to divide the words into pairs so that words in each pair have common prefix of length at least k.

Find the maximum possible value of k.

Input

The first line of input contains an even integer n — number of words ($1 \le n \le 2 \cdot 10^5$).

The following n lines contain words that the friends have. All words have same length. Total words length is less or equal to $2 \cdot 10^6$.

Output

Output maximal possible value of k.

Examples

standard input	standard output
4 aabc aacc bbbb bbbd	2
2 a b	0

Problem L. Shifting Roads

Time limit: 1 second
Memory limit: 512 megabytes

There are m asphalt-covered roads in the city, the i-th road is a segment between two points A_i and B_i with coordinates (x_i^A, y_i^A) and (x_i^B, y_i^B), respectively.

To test the new technology of shifting roads, the city administration can dismantle exactly one asphalt-covered road and build exactly one new asphalt-covered road with the asphalt obtained after dismantling, but the length of the new road should not exceed the length of the dismantled road.

After analyzing the current budget, the city administration understood that it would soon be able to serve exactly three asphalt-covered roads.

Therefore, a modernization program was proposed: the city administration wants to choose three roads that will remain covered with asphalt, and also, possibly, to shift one of them. All other asphalt-covered roads will have to be removed permanently. After that, the three remaining asphalt-covered roads must necessarily form a *connected* asphalt-covered area, that is, between any two asphalt-covered points it should be possible to drive by asphalt-covered roads.

Now the city administration wants to know: how many ways are there to choose three roads to complete the modernization program?

Input

The first line of input gives an integer m — the number of asphalt-covered roads in the city ($3 \le m \le 100$).

Then m lines are given. The i-th line contains four integers: x_i^A, y_i^A, x_i^B, y_i^B — coordinates of points A_i and B_i of the beginning and end of i-th road respectively.

All coordinates of points are integers and do not exceed 10^4 in absolute value. The endpoints of any road are different.

Output

On the only line of the output print a single integer k — the number of ways to choose three roads to complete the modernization program.

Example

standard input	standard output
4 1 1 2 3 1 3 2 1 3 1 4 3 2 6 3 6	3

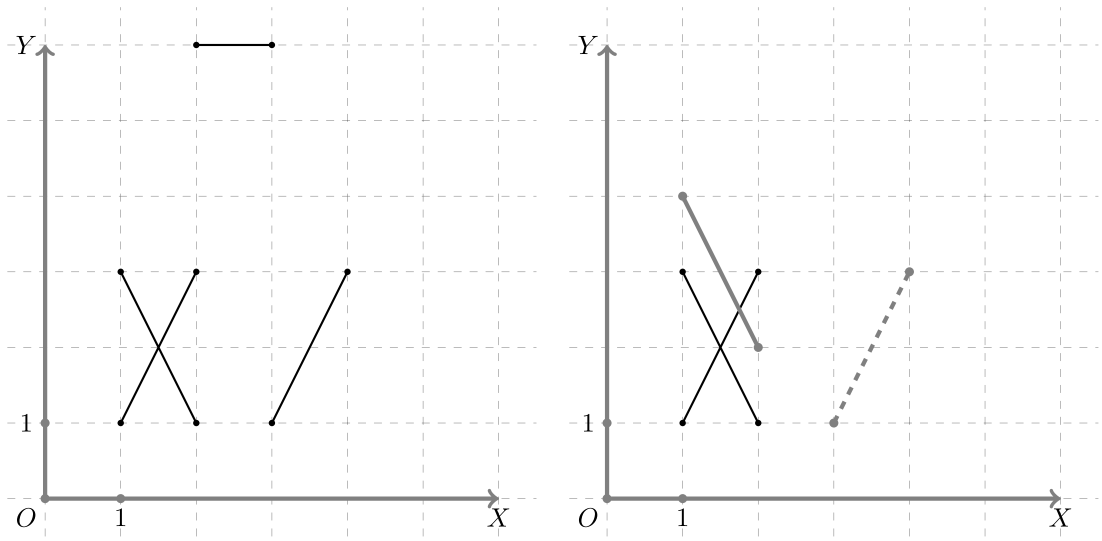
Note

In the example to complete the modernization program, you can choose three roads in one of three ways:

a) roads $\{1, 2, 3\}$ with coordinates $(1, 1) - (2, 3)$, $(1, 3) - (2, 1)$, $(3, 1) - (4, 3)$ respectively, and shift road 3 to new coordinates $(1, 4) - (2, 2)$

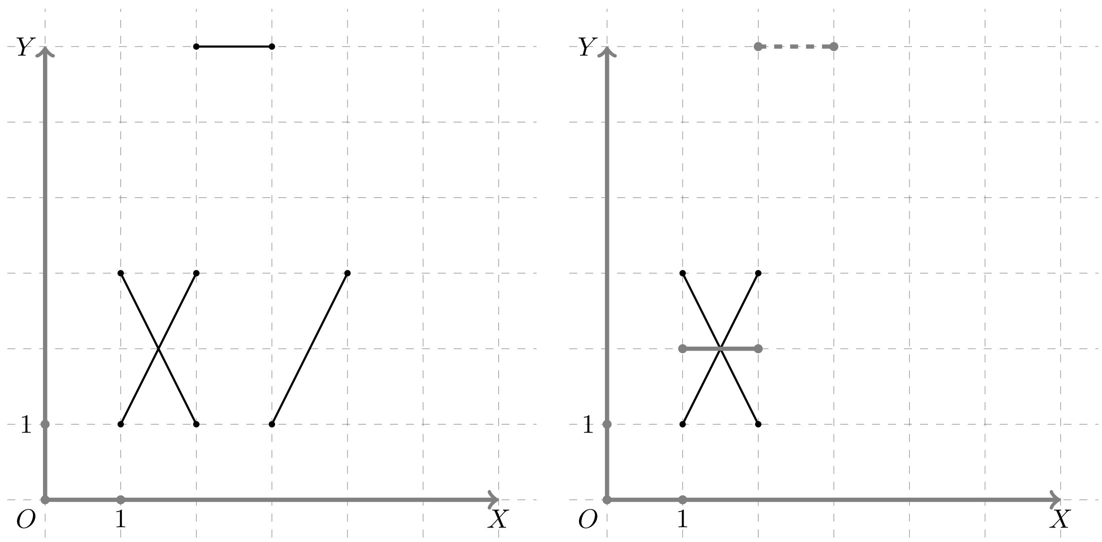
b) roads $\{1, 2, 4\}$ with coordinates $(1, 1) - (2, 3)$, $(1, 3) - (2, 1)$, $(2, 6) - (3, 6)$ respectively, and shift road 4 to new coordinates $(1, 2) - (2, 2)$

c) roads $\{2, 3, 4\}$ with coordinates $(1, 3) - (2, 1)$, $(3, 1) - (4, 3)$, $(2, 6) - (3, 6)$ respectively, and shift road 4 to new coordinates $(2, 1) - (3, 1)$



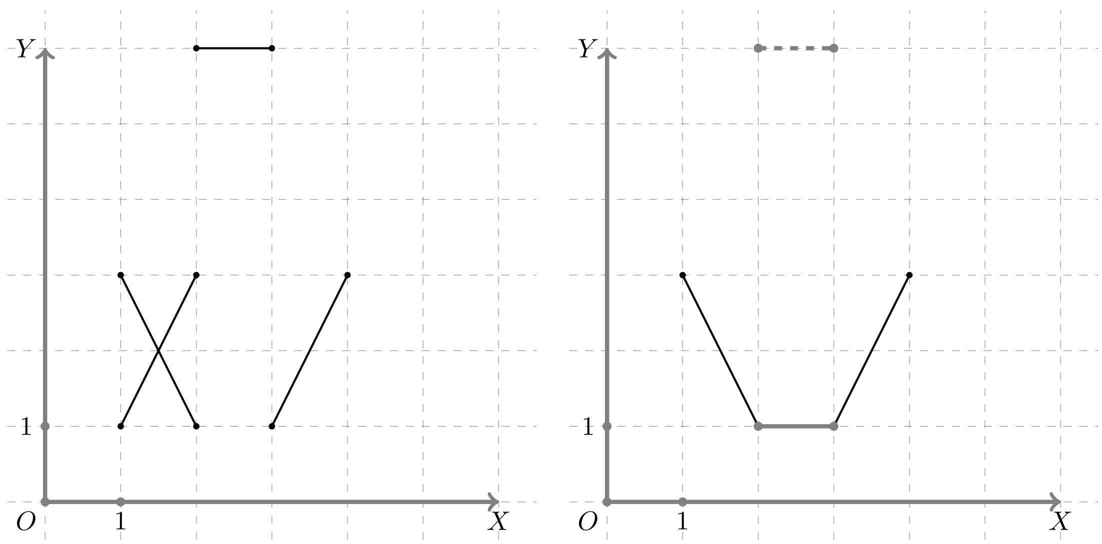
Picture for a)

Shifting the 3 road from coordinates $(3,1)-(4,3)$ to new coordinates $(1,4)-(2,2)$



Picture for b)

Shifting the 4 road from coordinates $(2,6)-(3,6)$ to new coordinates $(1,2)-(2,2)$



Picture for c)

Shifting the 4 road from coordinates $(2,6)-(3,6)$ to new coordinates $(2,1)-(3,1)$