

## La Salle-Pui Ching Programming Challenge 培正喇沙编程挑战赛 2017

### A. Ambiguous Dates

time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

There are two popular formats for representing a date: `day/month/year` or `month/day/year`. For example, today can be represented as 15/8/2017 or 8/15/2017.

Sometimes (like on today), using one way or another should pose no confusion — it is immediately understood that the date is the 15th of August. On other days, however, the two representations may be interpreted as two different valid dates. For example, the 7th of August may be misinterpreted as the 8th of July, since both can be represented as 7/8/2017 (or 8/7/2017).

We say a date  $(D, M, Y)$  is ambiguous if  $D/M/Y$  and  $M/D/Y$ , when both interpreted in the `day/month/year` format, are different valid dates. For example,  $(7, 8, 2017)$  and  $(8, 7, 2017)$  are ambiguous, while  $(15, 8, 2017)$  and  $(10, 10, 2017)$  are not.

The total number of ambiguous dates in the Gregorian calendar system on any given year is equal to  $12 \times 11 = 132$ .

Now, suppose that in a hypothetical calendar system, there are  $M$  months, where the  $i$ -th month has  $D[i]$  days, numbered from 1 to  $D[i]$ . Assume that there are no leap years.

You are to carry out a calendar reform, by shuffling the array  $D[]$ , and your target is to minimize the total number of ambiguous dates in a calendar year. Specifically, you want to find a permutation  $p[1], p[2], \dots, p[M]$  of integers  $1, 2, \dots, M$ , such that the new calendar system, where the  $i$ -th month has  $D[p[i]]$  days, has the minimal number of ambiguous dates. Output that minimal number.

#### Input

The first line of input consists of a single integer  $M$ , the number of months in the hypothetical calendar system.

The second line of input consists of  $M$  integers  $D[1], D[2], \dots, D[M]$ , the original number of days in the  $i$ -th month.

For all test cases,  $1 \leq M \leq 10^5$ ,  $1 \leq D[i] \leq 10^5$ .

#### Output

Output a single integer, the minimal number of ambiguous dates after the calendar reform.

#### Examples

|   |                      |
|---|----------------------|
| <b>input</b>                              | <a href="#">Copy</a> |
| 12<br>31 28 31 30 31 30 31 31 30 31 30 31 |                      |
| <b>output</b>                             |                      |
| 132                                       |                      |

|               |                      |
|---------------|----------------------|
| <b>input</b>  | <a href="#">Copy</a> |
| 3<br>5 1 1    |                      |
| <b>output</b> |                      |
| 0             |                      |

## B. Bacteria Experiment

time limit per test: 1.5 s

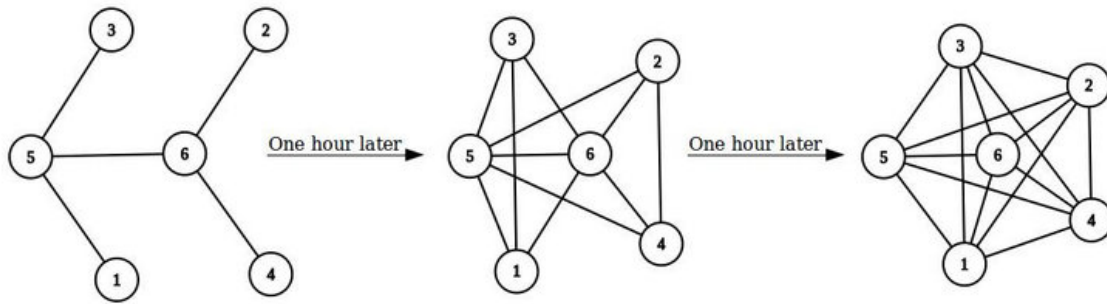
memory limit per test: 256 MB

input: standard input

output: standard output

A year after his bacteria experiment, Jason decided to perform another experiment on a new bacteria specie which evolves in a special way. The initial form of the bacteria can be considered as an undirected tree with  $N$  nodes in terms of graph theory. Every hour, an edge  $(x, y)$  is built if there exists a node  $z$  such that, in the previous hour, there exists edge  $(x, z)$  and edge  $(y, z)$ , but not edge  $(x, y)$ . The bacteria keep evolving until no more edges can be formed.

The following graph shows a type of bacteria which requires 2 hours to fully evolve:



As it may take months if not years for the bacteria to evolve to its ultimate form, it is impossible for Jason to stay at the laboratory to observe the change of the bacteria throughout the entire process. Therefore, he wants you to calculate the time required for the bacteria to fully evolve, so that he can just get back to the laboratory on time.

### Input

The first line contains an integer  $N$ . ( $2 \leq N \leq 5 \times 10^5$ )

The next  $N - 1$  line each contains two integers  $u$  and  $v$ , which means there exists an edge between node  $u$  and  $v$ . ( $1 \leq u, v \leq N$ )

The given graph is guaranteed to be a valid tree.

### Output

Output an integer, the time (in hours) required for the bacteria to fully evolve.

### Example

| input                              | Copy |
|------------------------------------|------|
| <pre> 6 1 5 5 3 5 6 6 2 6 4 </pre> |      |
| output                             |      |
| <pre> 2 </pre>                     |      |

## C. Cheering

time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

To boost contestants' performances in the 20th *La Salle - Pui Ching Programming Challenge*, the organizers have bought  $N$  robots to cheer for them. Each robot is supposed to display cheering slogans, letter by letter.

Unfortunately, due to some technical reasons, the display screen of each robot can only display one fixed character. Therefore, the organizers decided to arrange the robots in a row, thus forming a string of  $N$  letters (What a waste!). All letters are in uppercase.

The two hosting schools have abbreviated names `LSC` and `PCMS`, as we all know. Which of the two names appear (as a substring) in the string more often?

### Input

The first and only line of input consists of a string with  $N$  uppercase letters.

For all test cases,  $N \leq 100$ .

### Output

Let  $A$  be the number of occurrences of `LSC` in the given string.

Let  $B$  be the number of occurrences of `PCMS` in the given string.

If  $A > B$ , output `LSC`.

If  $A < B$ , output `PCMS`.

If  $A = B$ , output `Tie`.

### Examples

|                                   |                      |
|-----------------------------------|----------------------|
| <b>input</b>                      | <a href="#">Copy</a> |
| <code>GOLSC</code>                |                      |
| <b>output</b>                     |                      |
| <code>LSC</code>                  |                      |
| <b>input</b>                      | <a href="#">Copy</a> |
| <code>PCMSISTHEBEST</code>        |                      |
| <b>output</b>                     |                      |
| <code>PCMS</code>                 |                      |
| <b>input</b>                      | <a href="#">Copy</a> |
| <code>PCMSWILLNEVERBEATLSC</code> |                      |
| <b>output</b>                     |                      |
| <code>Tie</code>                  |                      |
| <b>input</b>                      | <a href="#">Copy</a> |
| <code>ADDOILEVERYONE</code>       |                      |
| <b>output</b>                     |                      |
| <code>Tie</code>                  |                      |

## D. Distribution of Days

time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

The Gregorian calendar is internationally the most widely used civil calendar. It is named after Pope Gregory XIII, who introduced it in October 1582.

In the Gregorian calendar, there are 28 days in February in a common year and 29 days in February in a leap year. Year  $Y$  is a leap year if and only if  $Y$  is a multiple of 400, or  $Y$  is a multiple of 4 and is not a multiple of 100.

Percy is curious about the distribution of days of the week of his birthday in his life. By checking the calendar, he quickly finds that in the years between 1999 and 2017 (inclusive), his birthday (in case you do not know, 27 February) appears only twice on both Tuesday and Thursday, three times on each of the other days of the week.

Percy finds counting the distribution of some days in some consecutive years really cool, so he decides to invent a way to quickly count the distribution.

Within 15 minutes, he successfully invented a fast program to do the calculation for years between 1583 and  $2 \times 10^9$ , inclusive. His program can answer 5000 queries in 1 second. However, he is not sure if the program works correctly, so he needs your help. Your task is simple, write your own program to do the calculation, so that Percy can check his program's correctness by comparing the outputs of different queries with your program.

In this problem, please assume the definition of leap years mentioned above is true for all years between 1583 and  $2 \times 10^9$ , inclusive.

### Input

The first line consists of a single integer,  $Q$ , denotes the number of queries. ( $1 \leq Q \leq 5000$ )

In the next  $Q$  lines, each describes a single query. The queries are in the format  $S \ E \ M \ D$ , which means you have to calculate the distribution of days of the week for the  $D$ -th day of the  $M$ -th month for all years between  $S$  and  $E$ , inclusive. ( $1583 \leq S \leq E \leq 2 \times 10^9$ , the days given are one of the 366 valid days)

### Output

Output  $Q$  lines, each answers a query given.

In each line output 7 integers, the frequencies of days of the weeks in this order: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday.

The order of answers should follow the order of queries given.

### Examples

|                     |             |
|---------------------|-------------|
| <b>input</b>        | <b>Copy</b> |
| 1<br>1999 2017 2 27 |             |
| <b>output</b>       |             |
| 3 3 2 3 2 3 3       |             |

|                                       |             |
|---------------------------------------|-------------|
| <b>input</b>                          | <b>Copy</b> |
| 2<br>2017 2017 8 15<br>2017 2021 2 29 |             |
| <b>output</b>                         |             |
| 0 0 1 0 0 0 0<br>0 0 0 0 0 0 1        |             |

|   |             |
|---|-------------|
| <b>input</b>  | <b>Copy</b> |
| 4<br>3141 5926 5 3<br>5897 9323 8 4<br>2718 2818 2 8<br>2222 2222 2 22                              |             |
| <b>output</b>   |             |
| 404 391 403 390 404 396 398<br>488 488 497 481 497 480 496<br>15 14 14 15 14 15 14<br>0 0 0 0 0 1 0 |             |

## E. Expected Score

time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

As the army of Byteland is getting stronger and stronger, Hackson, the King of Hackerland, decides to establish a better relationship with Byteland. The King of Byteland, who loves playing games, asked Hackson to play a game with him.

The rule of the game is as follows:

On the border between Byteland and Hackerland, there are  $2 \times N$  chests. The inner surface of each chest is marked with a number which ranges from 1 to  $N$ . The King of Byteland told Hackson that every number appears exactly two times, but since all the chests are closed, Hackson has no idea what numbers are marked on the inside of the chests.

Each round, Hackson can choose two different chests and open them at the same time. If the numbers marked on both chosen chests are the same, the score of this round will be equal to that number. Otherwise, the score of this round will be zero. Both chests, whether the marked numbers are the same or not, are then closed and remain in the game.

The game will last for  $R$  rounds, and **the final score of the game will be the highest score among all rounds**.

Currently,  $R - 1$  rounds have passed. Hackson would like to know, if he chooses two chests in the final round that maximizes his expected final score, what will his expected final score be?

### Input

The first line consists of two integers  $N$  and  $R$ . ( $1 \leq N, R \leq 10^5$ )

The next  $R - 1$  lines each consists of four integers  $x, y, C_x$  and  $C_y$ , which means that in the  $i$ -th round, the  $x$ -th and the  $y$ -th chest are chosen, and they are found out to be marked with numbers  $C_x$  and  $C_y$  respectively. ( $1 \leq x, y \leq 2N, x \neq y, 1 \leq C_x, C_y \leq N$ )

It is guaranteed that the given information is valid and consistent.

### Output

Output a number, the expected final score of Hackson if he plays wisely in the final round.

Your answer will be accepted if the relative error or absolute error, whichever is less, is not greater than  $10^{-6}$ .

### Examples

| input                     | Copy |
|---------------------------|------|
| 3 3<br>1 3 1 3<br>2 4 2 3 |      |
| output                    |      |
| 3                         |      |

| input  | Copy |
|--------|------|
| 2 1    |      |
| output |      |
| 0.5    |      |

| input          | Copy |
|----------------|------|
| 3 2<br>1 4 2 2 |      |
| output         |      |
| 2.166667       |      |

### Note

In the first sample, there are a total of 6 chests. The score for the first 2 rounds are 0 but since Hackson has already opened the two chests marked with number 3, the optimal strategy is to choose those two chests in the final round. Therefore the maximum expected final score is 3.

In the second sample, there are a total of 4 chests, so Hackson has no information about the chests, which means Hackson can only choose two chests at random. He has  $\frac{1}{6}$  chance of opening two chests marked with 1 and scoring 1 point,  $\frac{1}{6}$  chance of opening two chests marked with 2 and

scoring 2 points,  $\frac{2}{3}$  of choosing two chests marked with different numbers and scoring zero. Therefore, his expected final score is

$$\frac{1}{6} \times 1 + \frac{1}{6} \times 2 + \frac{2}{3} \times 0 = 0.5$$

In the third sample, there are a total of 6 chests. In round 1, Hackson scored 2 points. In the last round, Hackson's optimal strategy is to choose any two unknown chests. There is  $\frac{1}{6}$  chance that the two chests are marked with 3, which makes the final score 3. Therefore the maximum expected score is  $\frac{1}{6} \times 3 + \frac{5}{6} \times 2 = 2.16666 \dots$

## F. Frustrating Game

time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

The magician Alice is playing a game. Alice has  $N$  balls in a row, consists of red balls and blue balls. For convenience, the balls are numbered 1 to  $N$  from left to right. Her goal is to transform all balls into White by casting magic spells.

In a single spell, Alice can choose exactly  $R$  red balls and  $B$  blue balls at the same time, and then cast a spell on them, so all these  $R + B$  chosen balls will become white balls. However, there is a restriction: As some magical power may remains in the white balls, Alice cannot perform spells when there exists at least one white ball located between the leftmost chosen ball and the rightmost chosen ball.

For instance, if Alice balls 2, 6 and 7 have became white balls by previous spells, she cannot choose the set of balls  $\{3, 8, 9\}$  in a single spell as there exist a white ball numbered 5 between them. Similarly, single spell on balls  $\{1, 3, 4\}$ ,  $\{3, 4, 9\}$  or  $\{1, 9, 10\}$  are invalid, while single spell on balls  $\{3, 4, 5\}$  or  $\{8, 9, 10\}$  are valid.

As Alice has spent more than hundred of years to achieve the goal and end up in failures, she is very frustrated. So, she promised that she will teach her magic skills to the one who can solve this frustrating game.

Nothing is more appealing than having an opportunity to learn magic! So you are trying to help Alice to achieve the goal.

### Input

The first line contains 3 positive integers,  $N$ ,  $R$  and  $B$ . ( $R, B \geq 1$ ,  $1 \leq R + B \leq N \leq 10^5$ )

The second line contains a string  $S$ , consisting of  $N$  characters  $R$  (red) and  $B$  (blue). The  $i$ -th character is the color of the  $i$ -th ball from the left.

### Output

If it is impossible to achieve the goal, output NO on the first line.

Otherwise, output YES, followed by an integer  $S$  on the second line – the number of spells ( $S$ ) you would like to cast.

In the next  $S$  lines, each should indicate a spell with  $R + B$  integers, representing the indices of the balls you have selected for this spell. You can output the indices in arbitrary order within the same line.

Please note that the spells should be printed in their **chronological order**. If there are more than one solution, you can output any of them.

### Examples

|                               |                      |
|-------------------------------|----------------------|
| <b>input</b>                  | <a href="#">Copy</a> |
| 6 1 1<br>BBRBRR               |                      |
| <b>output</b>                 |                      |
| YES<br>3<br>1 6<br>2 5<br>3 4 |                      |

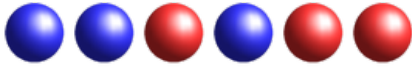
|                   |                      |
|-------------------|----------------------|
| <b>input</b>      | <a href="#">Copy</a> |
| 8 1 2<br>RBRBRBRB |                      |
| <b>output</b>     |                      |
| NO                |                      |

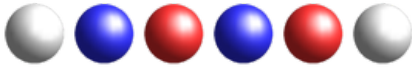
|                                     |                      |
|-------------------------------------|----------------------|
| <b>input</b>                        | <a href="#">Copy</a> |
| 10 2 3<br>BRRRBRRBB                 |                      |
| <b>output</b>                       |                      |
| YES<br>2<br>1 2 3 9 10<br>4 5 6 7 8 |                      |

### Note

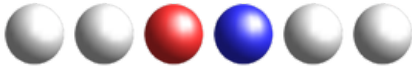
In the first sample, the initial state is as follow:



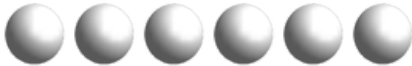
After casting the first spell:



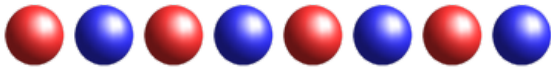
And after the second spell:



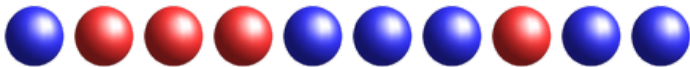
And the final state:



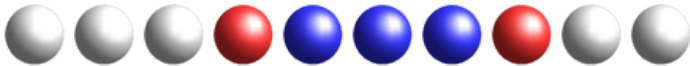
In the second sample, the initial state is as follow:



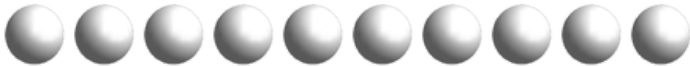
In the third sample, the initial state is as follow:



After casting the first spell:



And at last:





## G. Gravitational Tetris

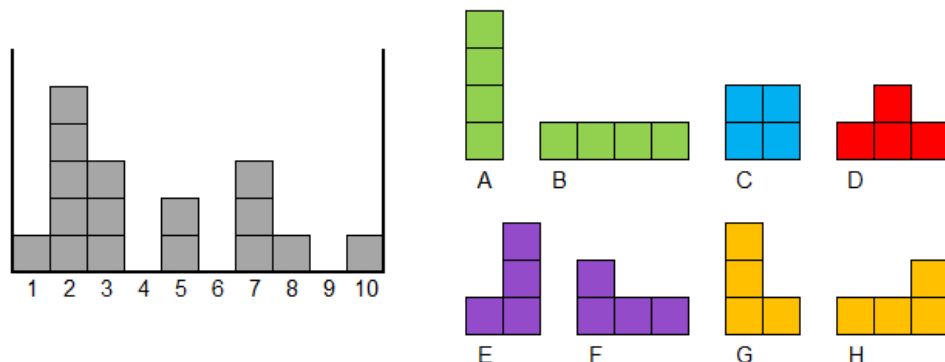
time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

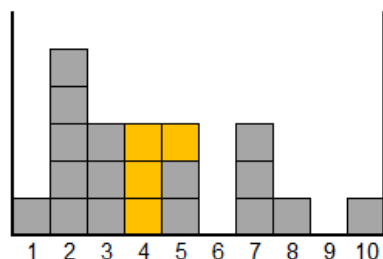
output: standard output

Gravitational Tetris is a Tetris variant that individual blocks can freely fall to the bottom. Therefore the game state of Gravitational Tetris can be described by an array of 10 integers, each representing the number of blocks (the height) of the column. The columns are numbered from 1 to 10. For example, the game state below can be described by the array  $[1, 5, 3, 0, 2, 0, 3, 1, 0, 1]$ .



You are given a valid Gravitation Tetris game state. Your task is to continue to play the game such that all blocks are eliminated. A row will be eliminated when there are 10 blocks in the row.

Since the blocks can freely fall, there are only 8 distinct Tetrominoes (shapes), with all rotations taken into account. The letter below each Tetromino denotes its type and its left-most column. In each move, you should select a Tetromino type and a column number  $c$ . For type A, it will fall onto column  $c$ . For type B, it will fall onto columns  $c, c+1, c+2$  and  $c+3$ . For type C, E and G, it will fall onto columns  $c$  and  $c+1$ . For type D, F and H, it will fall onto columns  $c, c+1$  and  $c+2$ . For example, from the above game state, after choosing type G and column 4, it will become:



You are allowed to use at most 1000 moves and there is no height restriction. Completed rows will be eliminated automatically.

### Input

The input consists of 10 integers,  $H_1, H_2, \dots, H_{10}$  – the height of the columns from left to right. It is guaranteed that it is a valid game state and a solution exists.

$0 \leq H_i \leq 12$ . At least one of them is 0 but their sum is not 0.

### Output

Output any sequence of moves that can eliminate all blocks. You are not required to minimize the number of moves.

In the first line output a single integer  $M$  – the number of moves.

In each of the next  $M$  lines output the type of Tetromino (A to H), a space, then the column number (1 to 10). Your answer will be regarded as incorrect if the column number is invalid for the type (e.g. column 8, 9 or 10 for type B).

### Examples

| input                  | Copy |
|------------------------|------|
| 1 1 0 1 1 1 0 1 0 2    |      |
| output                 |      |
| 3<br>B 4<br>H 1<br>H 7 |      |

input

Copy

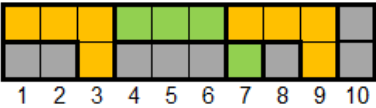
1 5 3 0 2 0 3 1 0 1

output

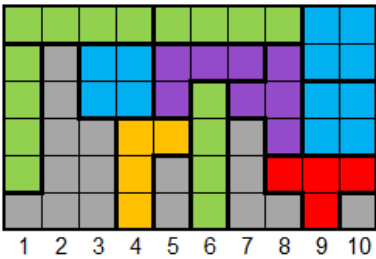
11  
G 4  
A 1  
A 6  
D 8  
C 3  
C 9  
C 9  
E 7  
F 5  
B 1  
B 5

Note

Explanation for Sample 1:



Explanation for Sample 2:



## H. Hit!

time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

"Hit!" is a popular game in ancient Byteland.

The very first version of the game is quite simple: each player picks up a stone and throws it at a circle drawn on the ground. A player wins if his/her stone lands inside the circle.

After 20 years of practice, Bitman, a young man living in ancient Byteland, has mastered the skill of throwing stones – he can throw a stone at any specific place he wants. With such skill, Bitman plays "Hit!" without losing a single game. He simply targets every stone at the center of the circle!

The King of Hackerland hears the story of Bitman and wants to challenge him with a harder, though still very simple, version of "Hit!".

In each game, two circles which share a positive common area are drawn on the ground. In order to win, the player must throw a stone at the common area of the two circles.

As Bitman had no idea how to target his stone at the common area, he asks for your help. Given the coordinates of the centers and radii of the two circles, please tell Bitman the coordinates of any point he can target at such that he can win the game.

For simplicity, you can consider the landing position of the stone as a single point.

### Input

The input consists of two lines, each describes one circle drawn on the ground. Each line contains three **integers**  $x$ ,  $y$  and  $r$ , denoting respectively the  $x$ -coordinate,  $y$ -coordinate, and the radius of a circle.

All coordinates have their absolute value no more than 100, and  $1 \leq r \leq 100$  for both circles.

### Output

Output two numbers, the  $x$ -coordinate and  $y$ -coordinate of a point where Bitman can throw his stone at to win the game.

Your answer will be accepted if for each of the two circles, the point lies inside the circle or that the distance between the point and the circle is not greater than  $10^{-5}$ .

### Examples

|                        |                      |
|------------------------|----------------------|
| <b>input</b>           | <a href="#">Copy</a> |
| <pre>0 0 3 3 4 3</pre> |                      |
| <b>output</b>          |                      |
| <pre>1.5 2.5</pre>     |                      |

|                            |                      |
|----------------------------|----------------------|
| <b>input</b>               | <a href="#">Copy</a> |
| <pre>-7 -9 3 -4 -4 5</pre> |                      |
| <b>output</b>              |                      |
| <pre>-6 -7</pre>           |                      |

### Note

In the first sample,  $(1.5, 2.5)$  is a possible answer as it lies inside the common area of two circles drawn. Please note that there exists more than one possible answer in this case. For example,  $(2, 2)$ ,  $(1, 2)$  and  $(2.1, 1.87)$  are also possible answers.

## I. Inverted Signs

time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

Arya lives in a magical world. You can view it as a number line.

There are  $N$  citizens in total. The  $i$ -th citizen have his own house in position  $i$  with height  $H_i$ . Noted that  $H_i$  could be non positive, which means their house is actually built beneath the horizon.

Arya is the hand of the king and she thinks that the buildings are a bit out of order. She denote the chaos index of the world as  $\sum_{i=1}^{N-1} |H_{i+1} - H_i|$ .

Noted that  $h_i$  could be equal to zero as the world is magical.

She feels that the current chaos index of the world is too high. So she designs to change the world a bit by using her super power.

She could flip a continuous sequence of building, i.e., she could choose two arbitrary integer  $L$  and  $R$ , where  $1 \leq L \leq R \leq N$ , and invert the signs of  $H_i$  for all  $L \leq i \leq R$  (positive to negative and vice versa).

As Arya is weak, she would only do this operation exactly once. Arya wants to minimize the chaos index after the operation. Being a good guy, help Arya to find the lowest possible chaos index after exactly flipping one continuous sequence of building.

### Input

The first line contains an integer  $N$ . ( $1 \leq N \leq 10^6$ )

The second line consist of  $N$  integers, the  $i$ -th integer is  $H_i$ . ( $-10^9 \leq H_i \leq 10^9$ )

### Output

Output consist only one integer in a single line, the lowest possible chaos index after exactly flipping one continuous sequence of building.

### Examples

|                |             |
|----------------|-------------|
| <b>input</b>   | <b>Copy</b> |
| 4<br>1 -2 -3 4 |             |
| <b>output</b>  |             |
| 3              |             |

|                   |             |
|-------------------|-------------|
| <b>input</b>      | <b>Copy</b> |
| 5<br>-3 -2 0 -5 3 |             |
| <b>output</b>     |             |
| 10                |             |

### Note

In the first sample, Arya should flip house 2 to 3, so that the heights become 1, 2, 3, 4 and the chaos index would be  $|2 - 1| + |3 - 2| + |4 - 3| = 3$ .

$|x - y|$  is the absolute difference between  $x$  and  $y$ .

## J. Juicy Candies

time limit per test: 2.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

Alex enjoys eating snacks, in particular candies. Recently he discovered a brand of berry-themed juicy candies, with three flavors: blueberry (B), raspberry (R), and strawberry (S). For brevity, we would, for example, use the term *B-candy* to refer to blueberry-flavored candy.

Alex has  $B$  B-candies,  $R$  R-candies, and  $S$  S-candies. He now wants to form sequences of candies, such that:

- All Alex's candies are used
- Any two adjacent candies have different flavors

For each candy sequence, there is a string of length  $(B + R + S)$  corresponding to it. If the  $i$ -th candy in the sequence is strawberry-flavored, for example, then the  $i$ -th character of the string would be S. Similarly for blueberry- and raspberry-flavored candies.

Before enjoying the candies, Alex would like to solve the following challenge: among all strings which corresponds to a candy sequence, what is the  $K$ -th lexicographically smallest string?

For two strings  $s[1...L]$  and  $t[1...L]$ ,  $s$  is said to be lexicographically smaller than  $t$ , if there exists an index  $i$  ( $1 \leq i \leq L$ ), such that  $s[j] = t[j]$  for  $1 \leq j < i$ , and  $s[i] < t[i]$ . We naturally use the character relation  $B < R < S$ .

### Input

The first and only line of input consists of four space-separated integers  $B$ ,  $R$ ,  $S$ , and  $K$ .

For all test cases,  $1 \leq B, R, S \leq 200$ ,  $1 \leq K \leq 10^{18}$ .

### Output

If the total number of strings is less than  $K$ , output `None`.

Otherwise, output the  $K$ -th lexicographically smallest string that corresponds to a candy sequence with  $B$  B-candies,  $R$  R-candies, and  $S$  S-candies.

### Examples

|               |                      |
|---------------|----------------------|
| <b>input</b>  | <a href="#">Copy</a> |
| 1 1 1 1       |                      |
| <b>output</b> |                      |
| BRS           |                      |

|               |                      |
|---------------|----------------------|
| <b>input</b>  | <a href="#">Copy</a> |
| 3 1 1 1       |                      |
| <b>output</b> |                      |
| BRBSB         |                      |

|               |                      |
|---------------|----------------------|
| <b>input</b>  | <a href="#">Copy</a> |
| 1 4 1 1       |                      |
| <b>output</b> |                      |
| None          |                      |

|               |                      |
|---------------|----------------------|
| <b>input</b>  | <a href="#">Copy</a> |
| 2 3 4 40      |                      |
| <b>output</b> |                      |
| SBRSRSBSR     |                      |

|                      |                      |
|----------------------|----------------------|
| <b>input</b>         | <a href="#">Copy</a> |
| 7 7 7 777            |                      |
| <b>output</b>        |                      |
| BRBRBSRSBSRSBSRSBSRS |                      |

|                              |                 |
|------------------------------|-----------------|
| <b>input</b>                 | <div>Copy</div> |
| <i>7 7 7 7777</i>            |                 |
| <b>output</b>                |                 |
| <i>BRBRBSRSRBRBRBSRSBSBS</i> |                 |

|                               |                 |
|-------------------------------|-----------------|
| <b>input</b>                  | <div>Copy</div> |
| <i>7 7 7 177777</i>           |                 |
| <b>output</b>                 |                 |
| <i>RSBRBSRSRBSRBRBSBSBSBR</i> |                 |

## K. Knights

time limit per test: 1.0 s

memory limit per test: 256 MB

input: standard input

output: standard output

The King of Byteland owns an army of well-trained knights. One day, he decides to attack Hackerland. Hackerland can be described as a grid with  $N$  rows and  $M$  columns. The cell on the  $i$ -th row,  $j$ -th column is called cell  $(i, j)$  for convenience. The north-west corner is  $(1, 1)$  while the south-east corner is  $(N, M)$ .

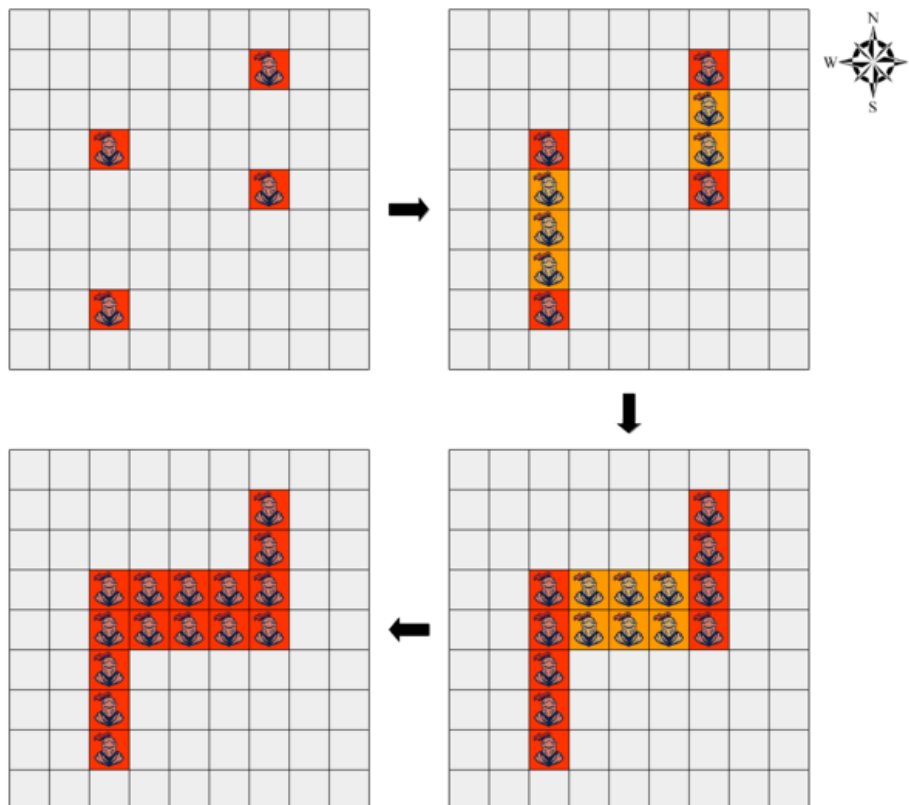
In order to win, the Byteland army must occupy all cells with knights. Now, the King of Byteland has carefully planned the attack. The attack can be divided into two phases.

In the first phase, a number of knights are sent to some specific cells of Hackerland.

After the first phase has ended, the second phase begins. In the second phase, The knights will start attacking the land of Hackerland. Their only way to attack is as follows: If at any moment, there are two knights on the same row or the same column, then all unoccupied cells between the two knights will be under attack, and more knights will be immediately sent to occupy these cells. This phase will end when no more cells can be occupied.

Please note that the knights are not allowed to move once they are sent to Hackerland.

In the following graph, the cells  $(2, 7)$ ,  $(4, 3)$ ,  $(5, 7)$  and  $(8, 3)$  are occupied in the first phase.



When the second phase starts, the cells between  $(4, 3)$  and  $(8, 3)$ ,  $(2, 7)$  and  $(5, 7)$  are under attack, and then occupied by other knights. Afterwards, the cells between  $(4, 3)$  and  $(4, 7)$ ,  $(5, 3)$  and  $(5, 7)$  will also be occupied. The second phase will end after this as no more cells can be occupied.

The attack is currently in the first phase.  $K$  cells of Hackerland are already occupied under the command of the King. You, the leader of the knights, are going to send some more knights (possibly none) to some cells of Hackerland of your choice before the second phase starts.

On one hand, you have to send enough knights to ensure that at the end of the war, all cells are occupied. On the other hand, you have to send as few knights as possible to minimize the chance of being discovered by the army of Hackerland. Your task is to find out the minimum number of knights to be sent.

### Input

The first line contains three integers:  $N$ ,  $M$  and  $K$ . ( $1 \leq N, M \leq 100$ ,  $0 \leq K \leq N \cdot M$ )

The next  $K$  lines each contains two integers:  $x$  and  $y$ , which means a knight was sent by the King to occupy cell  $(x, y)$ . ( $1 \leq x \leq N$ ,  $1 \leq y \leq M$ )

No any two knights are sent to the same cell.

**Output**

Output an integer, the minimum number of extra knights to be sent, on a single line.

**Examples**

|               |                      |
|---------------|----------------------|
| <b>input</b>  | <a href="#">Copy</a> |
| 1 1 0         |                      |
| <b>output</b> |                      |
| 1             |                      |

|               |                      |
|---------------|----------------------|
| <b>input</b>  | <a href="#">Copy</a> |
| 2 2 0         |                      |
| <b>output</b> |                      |
| 4             |                      |

|   |                      |
|---|----------------------|
| <b>input</b>  | <a href="#">Copy</a> |
| 3 3 8<br>1 1<br>1 2<br>1 3<br>2 1<br>2 3<br>3 1<br>3 2<br>3 3 |                      |
| <b>output</b>   |                      |
| 0   |                      |



## L. Let Me Count The Ways

time limit per test: 2.0 s

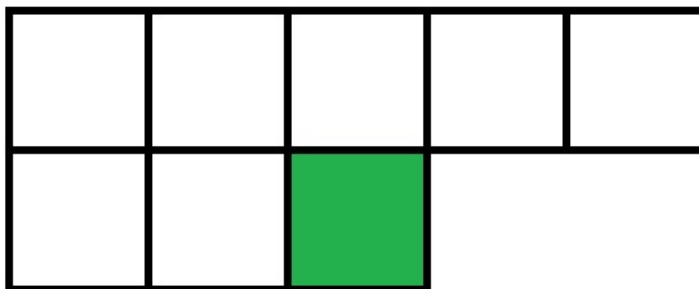
memory limit per test: 256 MB

input: standard input

output: standard output

You are given a grid with two rows. The first row has  $N$  cells and the second row has  $M$  cells. The cells are aligned to the left boundary of the grid. It is guaranteed that  $N \geq M$ , that is, the first row will not have fewer cells than the second row. By cell  $(R, C)$  we mean the cell which is on the  $R$ -th row and the  $C$ -th column.

For example, below is the grid corresponding to  $N = 5$ ,  $M = 3$ , and the coloured cell is  $(2, 3)$ :



You now want to fill the grid with integers  $1, 2, \dots, (N + M - 1), (N + M)$ , one integer per cell. A *valid assignment* is a way of filling the grid which satisfies the following conditions:

- Each of the integers  $1, 2, \dots, (N + M - 1), (N + M)$  appears exactly once;
- For every cell  $(R, C)$ , if the cell  $(R + 1, C)$  exists, then the number on cell  $(R, C)$  is greater than that on cell  $(R + 1, C)$ ; and
- For every cell  $(R, C)$ , if the cell  $(R, C + 1)$  exists, then the number on cell  $(R, C)$  is greater than that on cell  $(R, C + 1)$ .

Count the number of valid assignments, modulo  $1000000007 (= 10^9 + 7)$ . Note that  $1000000007$  is a prime number.

### Input

The first and only line of input consists of two space-separated integers  $N$  and  $M$ .

For all test cases,  $1 \leq M \leq N \leq 10^9$ .

### Output

Output the number of valid assignments, modulo  $1000000007$ .

### Examples

|               |             |
|---------------|-------------|
| <b>input</b>  | <b>Copy</b> |
| 5 3           |             |
| <b>output</b> |             |
| 28            |             |

|               |             |
|---------------|-------------|
| <b>input</b>  | <b>Copy</b> |
| 2 1           |             |
| <b>output</b> |             |
| 2             |             |

|               |             |
|---------------|-------------|
| <b>input</b>  | <b>Copy</b> |
| 4 4           |             |
| <b>output</b> |             |
| 14            |             |

|               |             |
|---------------|-------------|
| <b>input</b>  | <b>Copy</b> |
| 10 6          |             |
| <b>output</b> |             |
| 3640          |             |

**input**

Copy

123456789 23456890

**output**

484500998

**input**

Copy

987654321 896745320

**output**

0

---

[Codeforces](#) (c) Copyright 2010-2018 Mike Mirzayanov  
The only programming contests Web 2.0 platform