A. About Bacteria

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Qwerty the Ranger took up a government job and arrived on planet Mars. He should stay in the secret lab and conduct some experiments on bacteria that have funny and abnormal properties. The job isn't difficult, but the salary is high.

At the beginning of the first experiment there is a single bacterium in the test tube. Every second each bacterium in the test tube divides itself into k bacteria. After that some abnormal effects create b more bacteria in the test tube. Thus, if at the beginning of some second the test tube had x bacteria, then at the end of the second it will have $kx + b$ bacteria.

The experiment showed that after n seconds there were exactly z bacteria and the experiment ended at this point.

For the second experiment Qwerty is going to sterilize the test tube and put there t bacteria. He hasn't started the experiment yet but he already wonders, how many seconds he will need to grow at least z bacteria. The ranger thinks that the bacteria will divide by the same rule as in the first experiment.

Help Qwerty and find the minimum number of seconds needed to get a tube with at least z bacteria in the second experiment.

Input

The first line contains four space-separated integers k, b, n and t $(1 \le k, b, n, t \le 10^6)$ — the parameters of bacterial growth, the time Qwerty needed to grow z bacteria in the first experiment and the initial number of bacteria in the second experiment, correspondingly.

Output

Print a single number — the minimum number of seconds Qwerty needs to grow at least z bacteria in the tube.

Sample test(s)

input
```
3 1 3 5
```
output
```
2
```

input
```
1 4 4 7
```
output
```
3
```

input
```
2 2 4 100
```
output
```
0
```

B. Jumping on Walls

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Vasya plays a computer game with ninjas. At this stage Vasya's ninja should get out of a deep canyon.

The canyon consists of two vertical parallel walls, their height is n meters. Let's imagine that we split these walls into 1 meter-long areas and number them with positive integers from 1 to n from bottom to top. Some areas are safe and the ninja can climb them. Others are spiky and ninja can't be there. Let's call such areas *dangerous*.

Initially the ninja is on the lower area of the left wall. He can use each second to perform one of the following actions:

- climb one area up;
- climb one area down;
- jump to the opposite wall. That gets the ninja to the area that is exactly k meters higher than the area he jumped from. More formally, if before the jump the ninja is located at area x of one wall, then after the jump he is located at area $x + k$ of the other wall.

If at some point of time the ninja tries to get to an area with a number larger than n, then we can assume that the ninja got out of the canyon.

The canyon gets flooded and each second the water level raises one meter. Initially the water level is at the lower border of the first area. Ninja cannot be on the area covered by water. We can assume that the ninja and the water "move in turns" — first the ninja performs some action, then the water raises for one meter, then the ninja performs one more action and so on.

The level is considered completed if the ninja manages to get out of the canyon.

After several failed attempts Vasya started to doubt whether it is possible to complete the level at all. Help him answer the question.

Input

The first line contains two integers n and k $(1 \le n, k \le 10^5)$ — the height of the canyon and the height of ninja's jump, correspondingly.

The second line contains the description of the left wall — a string with the length of n characters. The i-th character represents the state of the i-th wall area: character "X" represents a dangerous area and character "-" represents a safe area.

The third line describes the right wall in the same format.

It is guaranteed that the first area of the left wall is not dangerous.

Output

Print "YES" (without the quotes) if the ninja can get out from the canyon, otherwise, print "NO" (without the quotes).

Sample test(s)

input

```
7 3
---X--X
-X--XX-
```

output

```
YES
```

input

```
6 2
--X-X-
X--XX-
```

output

```
NO
```

Note

In the first sample the ninja should first jump to the right wall, then go one meter down along the right wall, then jump to the left wall. The next jump can get the ninja from the canyon.

In the second sample there's no way the ninja can get out of the canyon.

C. Delivering Carcinogen

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Qwerty the Ranger arrived to the Diatar system with a very important task. He should deliver a special carcinogen for scientific research to planet Persephone. This is urgent, so Qwerty has to get to the planet as soon as possible. A lost day may fail negotiations as nobody is going to pay for an overdue carcinogen.

You can consider Qwerty's ship, the planet Persephone and the star Diatar points on a plane. Diatar is located in the origin of coordinate axes — at point $(0, 0)$. Persephone goes round Diatar along a circular orbit with radius R in the counter-clockwise direction at constant linear speed v_p (thus, for instance, a full circle around the star takes $\frac{2\pi R}{v_p}$ of time). At the initial moment of time Persephone is located at point (x_p, y_p).

At the initial moment of time Qwerty's ship is at point (x, y). Qwerty can move in any direction at speed of at most v ($v > v_p$). The star Diatar is hot (as all stars), so Qwerty can't get too close to it. The ship's metal sheathing melts at distance r ($r < R$) from the star.

Find the minimum time Qwerty needs to get the carcinogen to planet Persephone.

Input

The first line contains space-separated integers x_p, y_p and v_p ($-10^4 \le x_p, y_p \le 10^4$, $1 \le v_p < 10^4$) — Persephone's initial position and the speed at which it goes round Diatar.

The second line contains space-separated integers x, y, v and r ($-10^4 \le x, y \le 10^4$, $1 < v \le 10^4$, $1 \le r \le 10^4$) — The intial position of Qwerty's ship, its maximum speed and the minimum safe distance to star Diatar.

It is guaranteed that $r^2 < x^2 + y^2$, $r^2 < x_p^2 + y_p^2$ and $v_p < v$.

Output

Print a single real number — the minimum possible delivery time. The answer will be considered valid if its absolute or relative error does not exceed 10^{-6}.

Sample test(s)

input

```
10 0 1
-10 0 2 8
```

output

```
9.584544103
```

input

```
50 60 10
50 60 20 40
```

output

```
0.000000000
```

D. Cube Snake

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You've got an $n \times n \times n$ cube, split into unit cubes. Your task is to number all unit cubes in this cube with positive integers from 1 to n^3 so that:

- each number was used as a cube's number exactly once;
- for each $1 \le i < n^3$, unit cubes with numbers i and $i + 1$ were neighbouring (that is, shared a side);
- for each $1 \le i < n$ there were at least two different subcubes with sizes $i \times i \times i$, made from unit cubes, which are numbered with consecutive numbers. That is, there are such two numbers x and y, that the unit cubes of the first subcube are numbered by numbers $x, x + 1, ..., x + i^3 - 1$, and the unit cubes of the second subcube are numbered by numbers $y, y + 1, ..., y + i^3 - 1$.

Find and print the required numeration of unit cubes of the cube.

Input

The first line contains a single integer n $(1 \le n \le 50)$ — the size of the cube, whose unit cubes need to be numbered.

Output

Print all layers of the cube as n $n \times n$ matrices. Separate them with new lines. Print the layers in the order in which they follow in the cube. See the samples for clarifications.

It is guaranteed that there always is a solution that meets the conditions given in the problem statement.

Sample test(s)

input

```
3
```

output

```
1 4 17
2 3 18
27 26 19

8 5 16
7 6 15
24 25 20

9 12 13
10 11 14
23 22 21
```

Note

In the sample the cubes with sizes $2 \times 2 \times 2$ are numbered with integers $1, ..., 8$ and $5, ..., 12$.

E. Gripping Story

time limit per test: 4 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

One day Qwerty the Ranger witnessed two transport ships collide with each other. As a result, all contents of their cargo holds scattered around the space. And now Qwerty wants to pick as many lost items as possible to sell them later.

The thing is, both ships had lots of new gravitational grippers, transported to sale. A gripper is a device that can be installed on a spaceship and than draw items in space to itself ("grip") and transport them to the ship's cargo hold.

Overall the crashed ships lost n gravitational grippers: the i-th gripper is located at a point with coordinates (x_i, y_i). Each gripper has two features — p_i (the power) and r_i (the action radius) and can grip any items with mass of no more than p_i at distance no more than r_i. A gripper itself is an item, too and it has its mass of m_i.

Qwerty's ship is located at point (x, y) and has an old magnetic gripper installed, its characteristics are p and r. There are no other grippers in the ship's cargo holds.

Find the largest number of grippers Qwerty can get hold of. As he picks the items, he can arbitrarily install any gripper in the cargo hold of the ship, including the gripper he has just picked. At any moment of time the ship can have only one active gripper installed. We consider all items and the Qwerty's ship immobile when the ranger picks the items, except for when the gripper moves an item — then the item moves to the cargo holds and the ship still remains immobile. We can assume that the ship's cargo holds have enough room for all grippers. Qwerty can use any gripper he finds or the initial gripper an arbitrary number of times.

Input

The first line contains five integers x, y, p, r and n ($-10^9 \le x, y \le 10^9$, $1 \le p, r \le 10^9$, $1 \le n \le 250000$) — the ship's initial position, the initial gripper's features and the number of grippers that got into the space during the collision.

Next n lines contain the grippers' descriptions: the i-th line contains five integers x_i, y_i, m_i, p_i, r_i ($-10^9 \le x_i, y_i \le 10^9$, $1 \le m_i, p_i, r_i \le 10^9$) — the i-th gripper's coordinates and features.

It is guaranteed that all grippers are located at different points. No gripper is located at the same point with Qwerty's ship.

Output

Print a single number — the maximum number of grippers Qwerty can draw to his ship. You do not need to count the initial old magnet gripper.

Sample test(s)

input

```
0 0 5 10 5
5 4 7 11 5
-7 1 4 7 8
0 2 13 5 6
2 -3 9 3 4
13 5 1 9 9
```

output

```
3
```

Note

In the first sample you should get the second gripper, then use the second gripper to get the first one, then use the first gripper to get the fourth one. You cannot get neither the third gripper as it is too heavy, nor the fifth one as it is too far away.