Codeforces Round #190 (Div. 2)

A. Ciel and Dancing

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Fox Ciel and her friends are in a dancing room. There are n boys and m girls here, and they never danced before. There will be some songs, during each song, there must be exactly one boy and one girl are dancing. Besides, there is a special rule:

- either the boy in the dancing pair must dance for the first time (so, he didn't dance with anyone before);
- or the girl in the dancing pair must dance for the first time.

Help Fox Ciel to make a schedule that they can dance as many songs as possible.

Input
The first line contains two integers n and m $(1 \leq n, m \leq 100)$ — the number of boys and girls in the dancing room.

Output
In the first line print k — the number of songs during which they can dance. Then in the following k lines, print the indexes of boys and girls dancing during songs chronologically. You can assume that the boys are indexed from 1 to n, and the girls are indexed from 1 to m.

Sample test(s)

input
2 1

output
2 1 1 2 1

input
2 2

output
3 1 1 1 2 2 2

Note
In test case 1, there are 2 boys and 1 girl. We can have 2 dances: the 1st boy and 1st girl (during the first song), the 2nd boy and 1st girl (during the second song).

And in test case 2, we have 2 boys with 2 girls, the answer is 3.

B. Ciel and Flowers

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Fox Ciel has some flowers: r red flowers, g green flowers and b blue flowers. She wants to use these flowers to make several bouquets. There are 4 types of bouquets:

- To make a "red bouquet", it needs 3 red flowers.
- To make a "green bouquet", it needs 3 green flowers.
- To make a "blue bouquet", it needs 3 blue flowers.
- To make a "mixing bouquet", it needs 1 red, 1 green and 1 blue flower.

Help Fox Ciel to find the maximal number of bouquets she can make.

Input

The first line contains three integers r, g and b ($0 \le r, g, b \le 10^9$) — the number of red, green and blue flowers.

Output

Print the maximal number of bouquets Fox Ciel can make.

Sample test(s)

input
3 6 9
output
6

input
4 4 4
output
4

input
0 0 0
output
0

Note

In test case 1, we can make 1 red bouquet, 2 green bouquets and 3 blue bouquets.

In test case 2, we can make 1 red, 1 green, 1 blue and 1 mixing bouquet.

C. Ciel and Robot

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Fox Ciel has a robot on a 2D plane. Initially it is located in (0, 0). Fox Ciel code a command to it. The command was represented by string s. Each character of s is one move operation. There are four move operations at all:

- 'U': go up, (x, y) $\rightarrow$ (x, y+1);
- 'D': go down, (x, y) $\rightarrow$ (x, y-1);
- 'L': go left, (x, y) $\rightarrow$ (x-1, y);
- 'R': go right, (x, y) $\rightarrow$ (x+1, y).

The robot will do the operations in s from left to right, and repeat it infinite times. Help Fox Ciel to determine if after some steps the robot will located in (a, b).

Input

The first line contains two integers a and b, ($-10^9 \leq a, b \leq 10^9$). The second line contains a string s ($1 \leq |s| \leq 100$, s only contains characters 'U', 'D', 'L', 'R') — the command.

Output

Print "Yes" if the robot will be located at (a, b), and "No" otherwise.

Sample test(s)

input
2 2 RU
output
Yes

input
1 2 RU
output
No

input
-1 1000000000 LRRLU
output
Yes

input
0 0 D
output
Yes

Note

In the first and second test case, command string is "RU", so the robot will go right, then go up, then right, and then up and so on.

The locations of its moves are (0, 0) $\rightarrow$ (1, 0) $\rightarrow$ (1, 1) $\rightarrow$ (2, 1) $\rightarrow$ (2, 2) $\rightarrow$ ...

So it can reach (2, 2) but not (1, 2).

D. Ciel and Duel

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

Fox Ciel is playing a card game with her friend Jiro.

Jiro has n cards, each one has two attributes: $position$ (Attack or Defense) and $strength$. Fox Ciel has m cards, each one has these two attributes too. It's known that position of all Ciel's cards is Attack.

Now is Ciel's battle phase, Ciel can do the following operation many times:

1. Choose one of her cards X. This card mustn't be chosen before.
2. If Jiro has no alive cards at that moment, he gets the damage equal to (X's strength). Otherwise, Ciel needs to choose one Jiro's alive card Y, then:
   - If Y's position is Attack, then (X's strength) $\geq$ (Y's strength) must hold. After this attack, card Y dies, and Jiro gets the damage equal to (X's strength) - (Y's strength).
   - If Y's position is Defense, then (X's strength) $>$ (Y's strength) must hold. After this attack, card Y dies, but Jiro gets no damage.

Ciel can end her battle phase at any moment (so, she can use not all her cards). Help the Fox to calculate the maximal sum of damage Jiro can get.

Input

The first line contains two integers n and m ($1 \leq n, m \leq 100$) — the number of cards Jiro and Ciel have.

Each of the next n lines contains a string $position$ and an integer $strength$ ($0 \leq strength \leq 8000$) — the position and strength of Jiro's current card. Position is the string "ATK" for attack, and the string "DEF" for defense.

Each of the next m lines contains an integer $strength$ ($0 \leq strength \leq 8000$) — the strength of Ciel's current card.

Output

Output an integer: the maximal damage Jiro can get.

Sample test(s)

input

```
2 3
ATK 2000
DEF 1700
2500
2500
2500
```

output

```
3000
```

input

```
3 4
ATK 10
ATK 100
ATK 1000
1
11
101
1001
```

output

```
992
```

input

```
2 4
DEF 0
ATK 0
0
0
1
1
```

output

```
1
```

Note

In the first test case, Ciel has 3 cards with same $strength$. The best strategy is as follows. First she uses one of these 3 cards to attack "ATK 2000" card first, this attack destroys that card and Jiro gets 2500 - $2000 = 500$ damage. Then she uses the second card to destroy the "DEF 1700" card. Jiro doesn't get damage that time. Now Jiro has no cards so she can use the third card to attack and Jiro gets 2500 damage. So the answer is $500 + 2500 = 3000$.

In the second test case, she should use the "1001" card to attack the "ATK 100" card, then use the "101" card to attack the "ATK 10" card. Now Ciel still has cards but she can choose to end her battle phase. The total damage equals $(1001 - 100) + (101 - 10) = 992$.

In the third test case note that she can destroy the "ATK 0" card by a card with strength equal to 0, but she can't destroy a "DEF 0" card with that card.

E. Ciel the Commander

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Now Fox Ciel becomes a commander of Tree Land. Tree Land, like its name said, has n cities connected by $n - 1$ undirected roads, and for any two cities there always exists a path between them.

Fox Ciel needs to assign an officer to each city. Each officer has a rank — a letter from 'A' to 'Z'. So there will be 26 different ranks, and 'A' is the topmost, so 'Z' is the bottommost.

There are enough officers of each rank. But there is a special rule must obey: if x and y are two distinct cities and their officers have the same rank, then on the simple path between x and y there must be a city z that has an officer with higher rank. The rule guarantee that a communications between same rank officers will be monitored by higher rank officer.

Help Ciel to make a valid plan, and if it's impossible, output "Impossible!".

Input

The first line contains an integer n ($2 \le n \le 10^5$) — the number of cities in Tree Land.

Each of the following $n - 1$ lines contains two integers a and b ($1 \le a, b \le n, a \ne b$) — they mean that there will be an undirected road between a and b. Consider all the cities are numbered from 1 to n.

It guaranteed that the given graph will be a tree.

Output

If there is a valid plane, output n space-separated characters in a line — i-th character is the rank of officer in the city with number i.

Otherwise output "Impossible!".

Sample test(s)

input

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

output

```
A B B B
```

input

```
10
1 2
2 3
3 4
4 5
5 6
6 7
7 8
8 9
9 10
```

output

```
D C B A D C B D C D
```

Note

In the first example, for any two officers of rank 'B', an officer with rank 'A' will be on the path between them. So it is a valid solution.