

# Round Trip

You are playing Super Mario. Your character is currently on the left side of the river and want to go the right side to get some gold. After getting the gold, you want to go on the left side again. The distance from the left side to the right side is  $D$  units. In the river, there are two kinds of rocks, Big and Small. The big ones don't move no matter how many times Mario steps there, but the small ones will drown after Mario steps once. He can jump to any distance from 0 to  $D$  units to go from one rock to another. You have to find a way such that the maximum distance of a single jump is minimized.

## Input Format

Input will start with two numbers : **N and D**, where N is the number of rocks in the river and D is the Distance from the left side to the right side of the river.

- After that N lines follow, each of them representing one type of rock
  1. "R M" where R is the type of the rock ("S" small or "B" big) and M is the distance of the rock from the left side

## Constraints

- $N(0 \leq N \leq 100)$
- $D(0 < D < 1000000000)$ .

## Output Format

Output one number: the minimized maximum jump he can make.

## Sample Input 0

```
1 10
B 5
```

## Sample Output 0

```
5
```

## Sample Input 1

```
2 10
B 3
S 6
```

## Sample Output 1

```
7
```