C - Go Home

Time Limit: 2 sec / Memory Limit: 256 MiB

Score : 200 points

Problem Statement

There is a kangaroo at coordinate 0 on an infinite number line that runs from left to right, at time 0. During the period between time $i-1$ and time i, the kangaroo can either stay at his position, or perform a jump of length exactly i to the left or to the right. That is, if his coordinate at time $i-1$ is x, he can be at coordinate $x-i$, x or $x+i$ at time i. The kangaroo's nest is at coordinate X, and he wants to travel to coordinate X as fast as possible. Find the earliest possible time to reach coordinate X.

Constraints

- X is an integer.
- $1 \leq X \leq 10^9$

Input

The input is given from Standard Input in the following format:

$$X$$

Output

Print the earliest possible time for the kangaroo to reach coordinate X.

Sample Input 1

Copy

```
6
```

Sample Output 1

Copy

```
3
```

The kangaroo can reach his nest at time 3 by jumping to the right three times, which is the earliest possible time.

Sample Input 2

Copy

```
2
```

Sample Output 2

Copy

```
2
```

He can reach his nest at time 2 by staying at his position during the first second, and jumping to the right at the next second.

Sample Input 3

Copy

```
11
```

Sample Output 3

Copy

```
5
```

D - No Need

Time Limit: 2 sec / Memory Limit: 256 MiB

Score : 600 points

Problem Statement

AtCoDeer the deer has N cards with positive integers written on them. The number on the i-th card $(1 \leq i \leq N)$ is a_i. Because he loves big numbers, he calls a subset of the cards *good* when the sum of the numbers written on the cards in the subset, is K or greater.

Then, for each card i, he judges whether it is *unnecessary* or not, as follows:

- If, for any good subset of the cards containing card i, the set that can be obtained by eliminating card i from the subset is also good, card i is unnecessary.
- Otherwise, card i is NOT unnecessary.

Find the number of the unnecessary cards. Here, he judges each card independently, and he does not throw away cards that turn out to be unnecessary.

Constraints

- All input values are integers.
- $1 \leq N \leq 5000$
- $1 \leq K \leq 5000$
- $1 \leq a_i \leq 10^9 (1 \leq i \leq N)$

Partial Score

- 300 points will be awarded for passing the test set satisfying $N, K \leq 400$.

Input

The input is given from Standard Input in the following format:

$$\begin{array}{llll} N & K & & \\ a_1 & a_2 & \ldots & a_N \end{array}$$

Output

Print the number of the unnecessary cards.

Sample Input 1

Copy

```
3 6
1 4 3
```

Sample Output 1

Copy

```
1
```

There are two good sets: $\{2, 3\}$ and $\{1, 2, 3\}$.

Card 1 is only contained in $\{1, 2, 3\}$, and this set without card 1, $\{2, 3\}$, is also good. Thus, card 1 is unnecessary.

For card 2, a good set $\{2, 3\}$ without card 2, $\{3\}$, is not good. Thus, card 2 is NOT unnecessary.

Neither is card 3 for a similar reason, hence the answer is 1.

Sample Input 2

Copy

```
5 400
3 1 4 1 5
```

Sample Output 2 Copy

```
5
```

In this case, there is no good set. Therefore, all the cards are unnecessary.

Sample Input 3 Copy

```
6 20
10 4 3 10 25 2
```

Sample Output 3 Copy

```
3
```

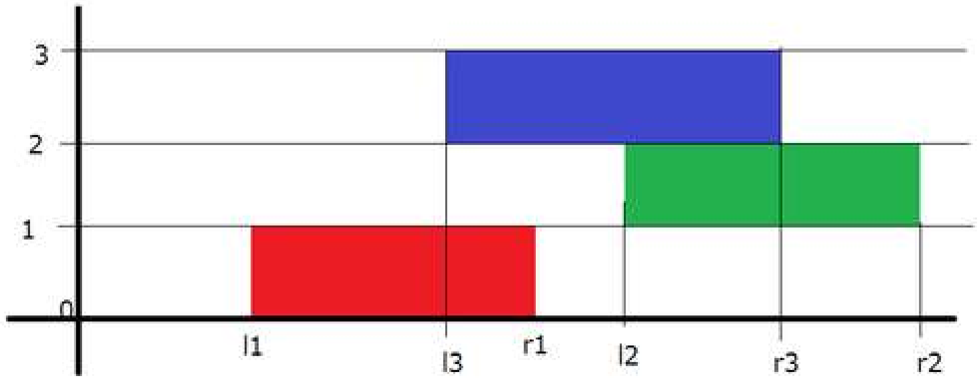
E - NarrowRectangles

Time Limit: 2 sec / Memory Limit: 256 MiB

Score : 1000 points

Problem Statement

AtCoDeer the deer found N rectangle lying on the table, each with height 1. If we consider the surface of the desk as a two-dimensional plane, the i-th rectangle $i(1 \leq i \leq N)$ covers the vertical range of $[i-1,i]$ and the horizontal range of $[l_i,r_i]$, as shown in the following figure:



AtCoDeer will move these rectangles horizontally so that all the rectangles are connected. For each rectangle, the cost to move it horizontally by a distance of x, is x. Find the minimum cost to achieve connectivity. It can be proved that this value is always an integer under the constraints of the problem.

Constraints

- All input values are integers.
- $1 \leq N \leq 10^5$
- $1 \leq l_i < r_i \leq 10^9$

Partial Score

- 300 points will be awarded for passing the test set satisfying $1 \leq N \leq 400$ and $1 \leq l_i < r_i \leq 400$.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \\
l_1 \quad r_1 \\
l_2 \quad r_2 \\
: \\
l_N \quad r_N
\end{array}
$$

Output

Print the minimum cost to achieve connectivity.

Sample Input 1

Copy

```
3
1 3
5 7
1 3
```

Sample Output 1

Copy

```
2
```

The second rectangle should be moved to the left by a distance of 2.

Sample Input 2

Copy

```
3
2 5
4 6
1 4
```

Sample Output 2 Copy

```
0
```

The rectangles are already connected, and thus no move is needed.

Sample Input 3 Copy

```
5
999999999 1000000000
1 2
314 315
500000 500001
999999999 1000000000
```

Sample Output 3 Copy

```
1999999680
```

Sample Input 4 Copy

```
5
123456 789012
123 456
12 345678901
123456 789012
1 23
```

Sample Output 4 Copy

```
246433
```

Sample Input 5 Copy

```
1
1 400
```

Sample Output 5 Copy

```
0
```

F - HonestOrUnkind

Time Limit: 2 sec / Memory Limit: 256 MiB

Score : 1300 points

Problem Statement

This is an interactive task.

AtCoDeer the deer came across N people. For convenience, the people are numbered 0 through $N-1$. Among them, A are *honest* and the remaining $B(=N-A)$ are *unkind*. All of these N people know who are honest and who are unkind, but AtCoDeer only knows that there are A honest and B unkind people. He is trying to identify all of the honest people by asking questions to these N people. For one question, AtCoDeer selects a and b $(0 \leq a, b \leq N-1)$, and asks person a the following question: "Is person b honest?"

An honest person will always answer correctly by "Yes" or "No". An unkind person, however, will answer by selecting "Yes" or "No" **arbitrarily**. That is, the algorithm used by an unkind person may not be simple one such as always lying or giving random fifty-fifty answers.

AtCoDeer can ask at most $2N$ questions. He will ask questions one by one, and the responses to the previous questions can be used when deciding the next question to ask.

Identify all of the honest people. If it is impossible (more formally, if, for any strategy of asking $2N$ questions, there exists a strategy for unkind people to answer the questions so that there are two or more possible sets of the honest people), report that fact.

Constraints

- $1 \leq A, B \leq 2000$

Input and Output

First, A and B are given from Standard Input in the following format:

A B

If identifying the honest people is impossible, the program must immediately print the following output and terminate itself:

```
Impossible
```

Otherwise, the program shall ask questions. Each question must be written to Standard Output in the following format:

`?` a b

Here, a and b must be integers between 0 and $N-1$ (inclusive). The response to the question will be given from Standard Input in the following format:

ans

Here, ans is either `Y` or `N`. `Y` represents "Yes"; `N` represents "No".

Finally, the answer must be written to Standard Output in the following format:

`!` $s_0 s_1 ... s_{N-1}$

Here, s_i must be `1` if person i is honest, and `0` if person i is unkind.

Judgement

- **After each output, you must flush Standard Output.** Otherwise you may get `TLE`.
- After you print the answer, the program must be terminated immediately. Otherwise, the behavior of the judge is undefined.
- When your output is invalid or incorrect, the behavior of the judge is undefined (it does not necessarily give `WA`).

Samples

In the following sample, $A = 2, B = 1$, and the answer is `101`.

Input	Output
$2\ 1$	
	$?\ 0\ 1$
N	
	$?\ 0\ 2$
Y	
	$?\ 1\ 0$
Y	
	$?\ 2\ 0$
Y	
	$?\ 2\ 2$
Y	
	$!\,101$

In the following sample, $A = 1, B = 2$, and the answer is `Impossible`.

Input	Output
$1\ 2$	
	$Impossible$