

Task Name

Full Timeline Coverage 1

Difficulty

Medium

Problem Description

Choose the fewest clips needed to cover the entire timeline from 0 to T, or report that it is impossible.

Problem Statement

You are given n clips. The i -th clip covers a time interval from $l[i]$ to $r[i]$. You may cut clips into smaller pieces if needed.

Your goal is to cover the entire timeline from 0 to T using as few clips as possible. If it is impossible to cover the whole interval $[0, T]$, print -1 .

Input Format

- The first line contains two integers n and T .
- Each of the next n lines contains two integers l and r .

Constraints

- $1 \leq n \leq 100$
- $1 \leq T \leq 100$
- $0 \leq l \leq r \leq 100$

Output Format

Print one integer: the minimum number of clips needed to cover $[0, T]$, or -1 if it cannot be done.

Example 1

Input

```
6 10 0 2 4 6 8 10 1 9 1 5 5 9
```

Output

```
3
```

Explanation

Take clips covering $[0, 2]$, $[1, 9]$, and $[8, 10]$. After cutting where needed, they cover the whole interval $[0, 10]$.

Example 2

Input

2 5 0 1 1 2

Output

-1

Explanation

The clips stop at time 2, so reaching time 5 is impossible.

Tags

Greedy, Intervals, Arrays

Reference Solution Idea

For each starting time, store the farthest ending time among clips that start there. Then greedily scan from left to right, always extending the current coverage as far as possible. This is the same idea as the classic minimum-jumps greedy strategy.

Complexity

- Time: $O(n + T)$
- Space: $O(T)$

Suggested testcase roles

- input00/output00 -> Sample, strength 0
- input01/output01 -> Sample, strength 0
- input02/output02 -> Hidden, strength 34
- input03/output03 -> Hidden, strength 33
- input04/output04 -> Hidden, strength 33