Problem E
Efficient Slabstones Rearrangement

Time limit: 2 seconds
Memory limit: 2048 megabytes

Problem Description

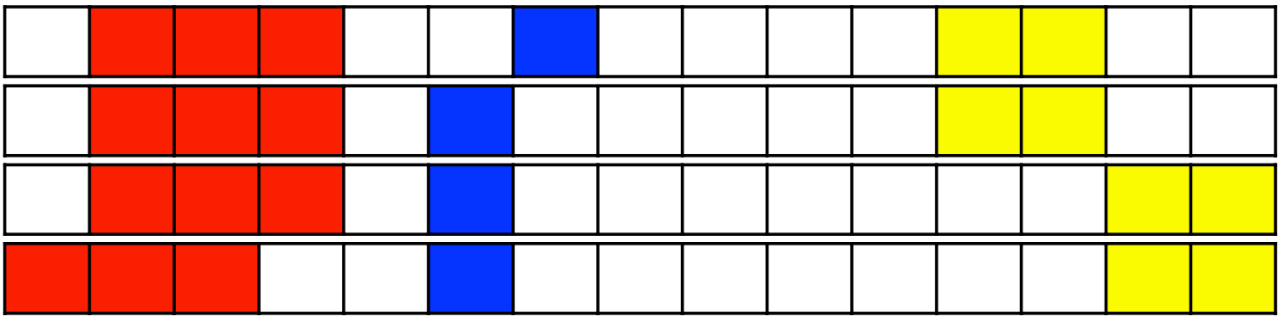
Barbara has a garden. The garden is long and narrow, divided into m equal-sized regions arranged in a row. Her friend, Babara, gave her n slabstones as birthday present. Barbara then placed these slabstones in her garden, so she can enjoy stepping slabstones from one to another every day. The i-th slabstone fully occupies the l_i-th to r_i-th region of the garden. The slabstones do not overlap, and any two slabstones have at least d empty regions between them.

Below is a valid placement of the slabstones with $m = 15$, $n = 3$, $d = 2$, and the three slabstones occupy the regions 2–4, 7–7, 12–13 respectively.



Barbara recently bought another slabstone that will occupy x consecutive regions in her garden. She will shift the original slabstones within the garden, then place the new slabstone somewhere in the garden. **After shifting** the original slabstones and placing the new slabstone, the slabstones cannot overlap, and any two slabstones must have at least d empty regions between them. The slabstones should remain non-overlapping **during slabstone rearrangement**.

Please note that, two slabstones can have less than d regions between them during slabstone rearrangement. For example, the following process is valid when $d = 2$:



Shifting a single slabstone to an adjacent region takes one minute. For example, the above rearrangement process takes 4 minutes. Now Barbara wants to know the minimum possible total time required to rearrange the slabstones, so she can save time for "other purposes".

Input Format

The first line contains four integers n, m, d and x. The i-th of the following n lines contains two integers l_i and r_i.

Output Format

The minimum possible total time (in minutes) to rearrange the slabstones so the new slabstone can be placed in the garden. If the new slabstone cannot be placed in the garden no matter how the slabstones are rearranged, just output `-1`.

Technical Specification

- $1 \leq n \leq 2000$
- $1 \leq d \leq m \leq 10^9$
- $1 \leq x \leq m \leq 10^9$
- $1 \leq l_i \leq r_i \leq m$ for $i \in \{1, 2, \ldots, n\}$
- $r_i + d + 1 \leq l_{i+1}$ for $i \in \{1, 2, \ldots, n-1\}$. That is, the slabstones are given in order from left to right.

Sample Input 1

```
3 15 2 3
2 4
7 7
12 13
```

Sample Output 1

```
4
```

Sample Input 2

```
5 100 1 75
2 3
5 7
11 13
17 19
23 29
```

Sample Output 2

```
9
```

Sample Input 3

```
1 100 99 1
1 1
```

Sample Output 3

```
-1
```