You are given an array tasks where tasks[i] = [$actual_i$, $minimum_i$]:

- $actual_i$ is the actual amount of energy you **spend to finish** the i^{th} task.
- $minimum_i$ is the minimum amount of energy you **require to begin** the i^{th} task.

For example, if the task is [10, 12] and your current energy is 11, you cannot start this task. However, if your current energy is 13, you can complete this task, and your energy will be 3 after finishing it.

You can finish the tasks in **any order** you like.

Return *the **minimum** initial amount of energy you will need to finish all the tasks*.

Example 1:

Input: tasks = [[1,2],[2,4],[4,8]]

Output: 8

Explanation:

Starting with 8 energy, we finish the tasks in the following order:

- 3rd task. Now energy = 8 - 4 = 4.
- 2nd task. Now energy = 4 - 2 = 2.
- 1st task. Now energy = 2 - 1 = 1.

Notice that even though we have leftover energy, starting with 7 energy does not work because we cannot do the 3rd task.

Example 2:

Input: tasks = [[1,3],[2,4],[10,11],[10,12],[8,9]]

Output: 32

Explanation:

Starting with 32 energy, we finish the tasks in the following order:

- 1st task. Now energy = 32 - 1 = 31.
- 2nd task. Now energy = 31 - 2 = 29.
- 3rd task. Now energy = 29 - 10 = 19.
- 4th task. Now energy = 19 - 10 = 9.
- 5th task. Now energy = 9 - 8 = 1.

Example 3:

Input: tasks = [[1,7],[2,8],[3,9],[4,10],[5,11],[6,12]]

Output: 27

Explanation:

Starting with 27 energy, we finish the tasks in the following order:

- 5th task. Now energy = 27 - 5 = 22.
- 2nd task. Now energy = 22 - 2 = 20.
- 3rd task. Now energy = 20 - 3 = 17.
- 1st task. Now energy = 17 - 1 = 16.
- 4th task. Now energy = 16 - 4 = 12.
- 6th task. Now energy = 12 - 6 = 6.

Constraints:

- $1 <= tasks.length <= 10^5$
- $1 <= actual_i <= minimum_i <= 10^4$