Problem 3683: Earliest Time to Finish One Task

Problem Information

Difficulty: Easy
Acceptance Rate: 83.69%
Paid Only: No
Tags: Array

Problem Description

You are given a 2D integer array `tasks` where `tasks[i] = [si, ti]`.

Each `[si, ti]` in `tasks` represents a task with start time `si` that takes `ti` units of time to finish.

Return the earliest time at which at least one task is finished.

Example 1:

Input: tasks = [[1,6],[2,3]]

Output: 5

Explanation:

The first task starts at time `t = 1` and finishes at time `1 + 6 = 7`. The second task finishes at time `2 + 3 = 5`. You can finish one task at time 5.

Example 2:

Input: tasks = [[100,100],[100,100],[100,100]]

Output: 200

Explanation:

All three tasks finish at time `100 + 100 = 200`.

Constraints:

* `1 <= tasks.length <= 100` * `tasks[i] = [si, ti]` * `1 <= si, ti <= 100`

Code Snippets

C++:

```
class Solution {
public:
int earliestTime(vector<vector<int>>& tasks) {

}
};
```

Java:

```
class Solution {
public int earliestTime(int[][] tasks) {

}
}
```

Python3:

```
class Solution:
def earliestTime(self, tasks: List[List[int]]) -> int:
```