Problem 2365: Task Scheduler II

Problem Information

Difficulty: Medium
Acceptance Rate: 54.50%
Paid Only: No
Tags: Array, Hash Table, Simulation

Problem Description

You are given a **0-indexed** array of positive integers `tasks`, representing tasks that need to be completed **in order** , where `tasks[i]` represents the **type** of the `ith` task.

You are also given a positive integer `space`, which represents the **minimum** number of days that must pass **after** the completion of a task before another task of the **same** type can be performed.

Each day, until all tasks have been completed, you must either:

* Complete the next task from `tasks`, or * Take a break.

Return _the**minimum** number of days needed to complete all tasks_.

Example 1:

Input: tasks = [1,2,1,2,3,1], space = 3 **Output:** 9 **Explanation:** One way to complete all tasks in 9 days is as follows: Day 1: Complete the 0th task. Day 2: Complete the 1st task. Day 3: Take a break. Day 4: Take a break. Day 5: Complete the 2nd task. Day 6: Complete the 3rd task. Day 7: Take a break. Day 8: Complete the 4th task. Day 9: Complete the 5th task. It can be shown that the tasks cannot be completed in less than 9 days.

Example 2:

Input: tasks = [5,8,8,5], space = 2 **Output:** 6 **Explanation:** One way to complete all tasks in 6 days is as follows: Day 1: Complete the 0th task. Day 2: Complete the 1st task. Day 3: Take a break. Day 4: Take a break. Day 5: Complete the 2nd task. Day 6: Complete the 3rd

task. It can be shown that the tasks cannot be completed in less than 6 days.

Constraints:

* `1 <= tasks.length <= 105` * `1 <= tasks[i] <= 109` * `1 <= space <= tasks.length`

Code Snippets

C++:

```
class Solution {
public:
long long taskSchedulerII(vector<int>& tasks, int space) {

}
};
```

Java:

```
class Solution {
public long taskSchedulerII(int[] tasks, int space) {

}
}
```

Python3:

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