

# Problem 621: Task Scheduler

## Problem Information

**Difficulty:** Medium

**Acceptance Rate:** 62.27%

**Paid Only:** No

**Tags:** Array, Hash Table, Greedy, Sorting, Heap (Priority Queue), Counting

## Problem Description

You are given an array of CPU `tasks`, each labeled with a letter from A to Z, and a number `n`. Each CPU interval can be idle or allow the completion of one task. Tasks can be completed in any order, but there's a constraint: there has to be a gap of \*\*at least\*\* `n` intervals between two tasks with the same label.

Return the \*\*minimum\*\* number of CPU intervals required to complete all tasks.

**Example 1:**

**Input:** tasks = ["A", "A", "A", "B", "B", "B"], n = 2

**Output:** 8

**Explanation:** A possible sequence is: A -> B -> idle -> A -> B -> idle -> A -> B.

After completing task A, you must wait two intervals before doing A again. The same applies to task B. In the 3rd interval, neither A nor B can be done, so you idle. By the 4th interval, you can do A again as 2 intervals have passed.

**Example 2:**

**Input:** tasks = ["A", "C", "A", "B", "D", "B"], n = 1

**Output:** 6

**Explanation:** A possible sequence is: A -> B -> C -> D -> A -> B.

With a cooling interval of 1, you can repeat a task after just one other task.

**Example 3:**

**Input:** tasks = ["A", "A", "A", "B", "B", "B"], n = 3

**Output:** 10

**Explanation:** A possible sequence is: A -> B -> idle -> idle -> A -> B -> idle -> idle -> A -> B.

There are only two types of tasks, A and B, which need to be separated by 3 intervals. This leads to idling twice between repetitions of these tasks.

**Constraints:**

\* `1 <= tasks.length <= 104` \* `tasks[i]` is an uppercase English letter. \* `0 <= n <= 100`

## Code Snippets

**C++:**

```
class Solution {
public:
    int leastInterval(vector<char>& tasks, int n) {
    }
};
```

**Java:**

```
class Solution {
public int leastInterval(char[] tasks, int n) {
    }
}
```

**Python3:**

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