

## Task scheduler (leetcode)

Task scheduler problem from leetcode which again is more of math than algorithms.

### Task Scheduler — LeetCode

Given a character array `tasks`, representing the tasks a CPU needs to do, where each letter represents a different task. Tasks could be done in any order. Each task is done in one unit of time. For each unit of time, the CPU could complete either one task or just be idle. However, there is a non-negative integer `n` that represents the cooldown period between two **same tasks** (the same letter in the array), that is that there must be at least `n` units of time between any two same tasks.

Return the least number of units of times that the CPU will take to finish all the given tasks.

The trick here is that scheduler doesn't run the same job again until n units have passed since it ran the same job. So, if we think which job has the most impact of this restriction, we will naturally get the answer that most frequent job will have most impact.

Now let's see what it will take to schedule most frequent job and forget about others. Let's take a sample: [A, A, A, B, B, C] and cool down time is n = 2. The way to schedule just A would be below, where blank space is idle unit.

A \_ \_ A \_ \_ A. We need 2 idle units and 1 unit for A for each (f-1) A, where f is frequency of A + another 1 unit for last A. So, we found a formula to find number of units needed for most frequent job: (n+1)\*(f-1) + 1.

Now let's say there are two elements which are most frequent [A, A, A, B, B, B, C] and n=2. Now we have 3 Bs as well. As you can see that we can just fit the B in existing empty space. And the only extra unit will be needed for last B which is placed at the end next to last A. So, if there are two most frequent element the answer would be: (n+1)\*(f-1) + 2.

Generalizing this if we have X number of most frequent jobs then answer would be (n+1)\*(f-1) + X.

So, these are the number of units needed for most frequent jobs, how about the other jobs that need to be scheduled which are not most frequent. We have to schedule those jobs as well, right. Since, we have some empty space in in each block, we can just place the new jobs in those spaces. And since there frequency is less than they will fit in empty space and won't go beyond last space.

Take an example: [A, A, A, B, B, C] can be scheduled as below (Bold ones are the job that is put on empty spaces) **A B C A B** \_ \_ A.

In above example we are able to schedule other job in the same empty space and didn't need a new unit. What if we have one more C. That can be fit in another empty space.

But what if we have two other jobs D and E. Yes, this needs a bit of thinking. If after scheduling most frequent ones and then filling the empty space with other jobs, there are jobs left then we can always extend the block with extra space. Let's see for example [A, A, A, B, B, C, C, D, E]

After scheduling most frequent one and then filling B and C, scheduling will look like **A B C A B C A**. To put D and E we will extend it like below.

**A B C \_ A B C \_ A**

And put D and E on those empty space. **A B C D A B C E A**

Which means that either after scheduling we have some empty spaces left **OR** all empty spaces are filled and some other empty spaces are created and filled. So how many units are needed for later case: Easy. As many units as there are jobs because there are no empty spaces.

So final answer is:  $\max((n+1)*(f-1) + X, \text{TotalJobs})$

Below simple C++ programs shows it:

```
int leastInterval(vector<char>& t, int n)
{
    int f = 0;
    int X = 0;
    int totalJobs = t.size();
    for (char c : t)
    {
        _m[c]++;
        if (f < _m[c])
            f = _m[c];
    }

    // Find how many elements have max frequency.
    for (auto a : _m)
    {
        if (a.second == f)
        {
            X++;
        }
    }

    return std::max((f - 1)*(n+1) + X , totalJobs);
}
```

Task Scheduler

Leetcode Medium

👍 3

💬

📄

🔖

More from Satyendra Mishra

Follow

✉

Feb 27, 2022

**Maximum Swap (LeetCode:670)**

This post discusses another Leet code question: Given a num swap any two digits at most once to get maximum number. Like the question in a previous post, there is a bit of math involved in this question as well. Maximum Swap — LeetCode Mathematically, the bigger the digits on...

Leetcode Medium

4 min read

🔖

Share your ideas with millions of readers. 

Write on Medium

Mar 6, 2022

**Insert into sorted circular linked list**

Another leet code question, it does look like a simple question but if you are not careful about all the corner cases then you may get surprised. Insert into a Sorted Circular Linked List — LeetCode One trick here is that given head node may not be the start of...

Leetcode Medium

3 min read

🔖

Mar 8, 2022

**How to escalate Air India complaints**

Sometime back I wrote about Air India's poor services for international flights. You can read that story here: Why does Air India offer poor services with higher ticket prices? Recently I ran into another customer support hell that Air India offers, this article talks about that. Good par...

Air India

3 min read

🔖

Feb 7, 2022

**Continuous Sub Array Sum (Leet Code)**

Pretty much every software engineer who wants to enhance their problem-solving skills for interviews (yes there is a difference between interviews these days and problem solving at work daily) looks into problems at Leet Code. I was looking at this problem and it is more of a math'...

Leetcode Medium

3 min read

🔖

Jan 23, 2022

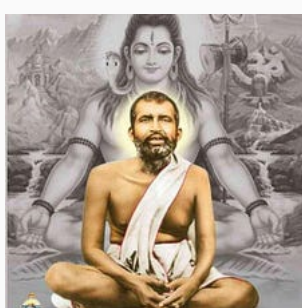
**Meditation and God**

Meditation has its deep roots in India, it started from India and slowly world adopted it. Today it is present in all the countries in one or other forms. Some just use the word Yoga and treat it differently from...

Meditation

2 min read

🔖



S

Satyendra Mishra

5 Followers

Follow

✉

#### More from Medium

- Santal Tech

No More Leetcode: The Stripe Interview Experience
- Naina Chaturvedi In Coders Mojo

Day 10 of 30 days of Data Structures and Algorithms and System Design Simplifie...
- Santal Tech

Every resource I used to get >\$500k software engineering offers
- Paul Simpson

Classification Model Accuracy Metrics, Confusion Matrix—and Thresholds!