Making Bank – AIO 2023

For this problem we can use a **greedy algorithm** to approach a solution.

For those who don't know, a greedy algorithm is one that makes the locally optimal choice at any given step. If a choice currently maximizes the output then it is almost always taken. These algorithms are really easy to implement as you only have to consider what's best for you in the moment.

Note: greedy algorithms don't always work, and that's why you should test your approach before implementing it

To implement the greedy algorithm here we simply check, on C days, whether we should attend the class that day, or make money, depending on which one maximizes the money by retirement.

Here's the implementation:

We initialize $skill = 1$ and $money = 0$. Iterate through each day,

If the day is a mandatory working day then we simply add our current skill level to money:

- $money = money + skill$

If the day is a class day, then you have 2 choices:

1. Go to class and earn skill, $skill = skill + 1$, this sacrifices any potential money you could have earned that day
2. Paint, and earn money, sacrificing some skill you could have gained: $money = money + skill$

From here we have to make the choice on which is better. One observation that can be made is the fact that, at some point, you will have to stop learning and just start earning. Why is this? The idea is that, at some point, trying to gain more skill leads to a diminishing return where there's no point in trying to gain anymore skill as it does not lead to increased gains.

With this reasoning we can simplify our approach to simply finding when the right time is to stop learning and start earning. We can set this out mathematically:

$$(skill + 1) \times remaining\ days > skill \times (remaining\ days + 1)?$$

On the left hand side, we are calculating the money **that can be earnt in the remaining days** if we learn today and start earning from tomorrow onwards. On the right hand side we are calculating whether its just suitable to start earning from today onwards. Based on which one is bigger we can make our choice.

Doing some algebra on this expression can simplify it down to just checking:

$$remaining\ days > skill?$$

But from here it's not necessary.

The solution in code is given below:

```python
N = int(input())
days = input()
skill = 1
money = 0
for i in range(N):
    if days[i] == 'M':
        money += skill # forced painting day
    else:
        remaining_days = N - i - 1
        if remaining_days > skill: # its better to learn today if this is true
            skill += 1
        else: # its better to earn from now on
            money += skill
print(money)
```