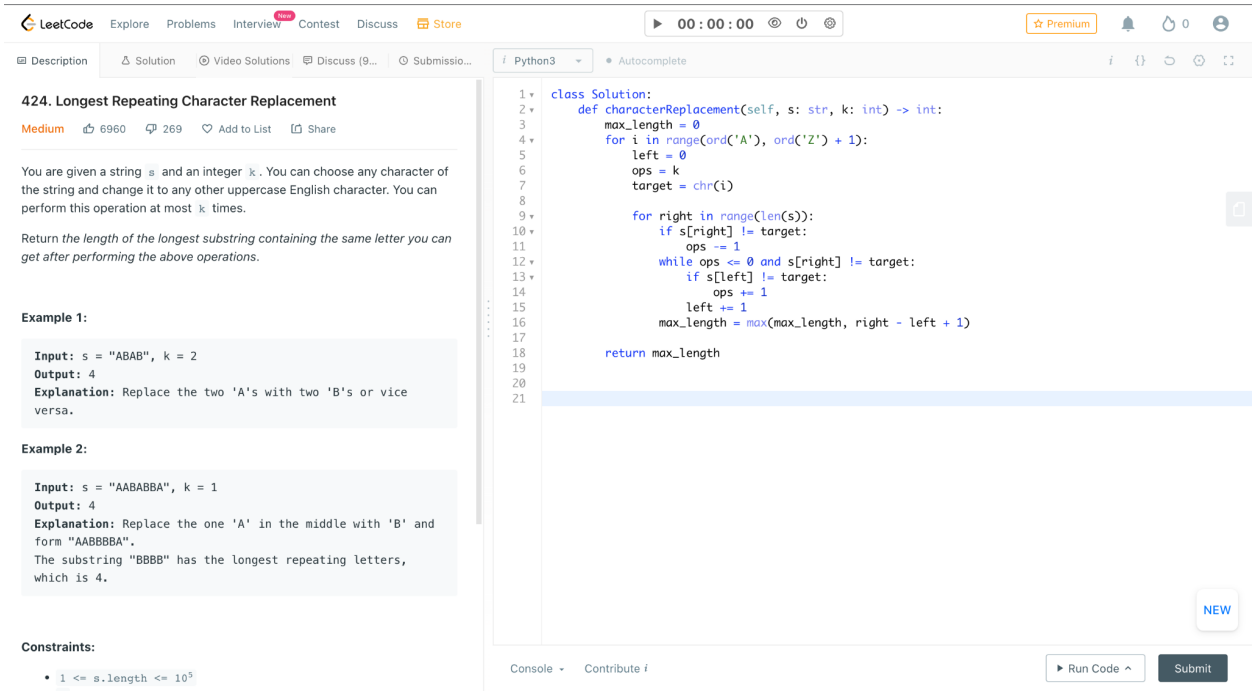
```python
class Solution:
    def numSubarrayProductLessThanK(self, nums: List[int], k: int) -> int:
        i, j = 0, 0
        product = 1
        subarrays = 0

        while j < len(nums):
            if j < len(nums) - 1 and product * nums[j] < k:
                product *= nums[j]
                j += 1
            elif i < j:
                subarrays += j - i
                product //= nums[i]
                i += 1
            else:
                i += 1
                j += 1

        return subarrays
```



```python
class Solution:
    def romanToInt(self, s: str) -> int:
        _dict = {
            'I' : 1,
            'V' : 5,
            'X' : 10,
            'L' : 50,
            'C' : 100,
            'D' : 500,
            'M' : 1000
        }

        subtract = {
            'I' : set(['V', 'X']),
            'X' : set(['L' , 'C']),
            'C' : set(['D', 'M'])
        }
        ans = 0
        i = 0
        while i < (len(s)):
            ch = s[i]
            if ch in subtract and i < len(s) - 1 and \
                s[i + 1] in subtract[ch]:
                ans += (_dict[s[i + 1]] - _dict[ch])
            else:
                ans += _dict[ch]
            i += 1
        return ans
```