12. Integer to Roman

Difficulty	medium
☑	☑
Finished	@July 19, 2023
Problem	strings
Previously asked company	Amazon Goldman Sachs
website	leetcode

Question:

Roman numerals are represented by seven different symbols: `I`, `V`, `X`, `L`, `C`, `D` and `M`.

```
SymbolValue
I             1
V             5
X             10
L             50
C             100
D             500
M             1000
```

For example, `2` is written as `II` in Roman numeral, just two one's added together. `12` is written as `XII`, which is simply `X + II`. The number `27` is written as `XXVII`, which is `XX + V + II`.

Roman numerals are usually written largest to smallest from left to right. However, the numeral for four is not `IIII`. Instead, the number four is written as `IV`. Because the one is before the five we subtract it making four. The same principle applies to the number nine, which is written as `IX`. There are six instances where subtraction is used:

- `I` can be placed before `V` (5) and `X` (10) to make 4 and 9.
- `X` can be placed before `L` (50) and `C` (100) to make 40 and 90.
- `C` can be placed before `D` (500) and `M` (1000) to make 400 and 900.

Given an integer, convert it to a roman numeral.

Example 1:

```
Input: num = 3
Output: "III"
Explanation: 3 is represented as 3 ones.
```

Example 2:

```
Input: num = 58
Output: "LVIII"
```

```
Explanation: L = 50, V = 5, III = 3.
```

Example 3:

```
Input: num = 1994
Output: "MCMXCIV"
Explanation: M = 1000, CM = 900, XC = 90 and IV = 4.
```

Optimal solution:

Time complexity: O(n)

Space complexity: O(n)

```
class Solution(object):
    def intToRoman(self, num):
        symList = [['I', 1], ['IV', 4], ['V', 5], ['IX', 9],
                   ['X', 10], ['XL', 40], ['L', 50], ['XC', 90],
                   ['C', 100], ['CD', 400], ['D', 500], ['CM', 900], ['M', 1000]]

        res = ''
        for sym, val in reversed(symList):
            if num // val:
                count = num // val
                res += (sym * count)
                num = num % val

        return res
```