



链滴

# [每日 LeetCode] 13. Roman to Integer

作者: [Hanseltu](#)

原文链接: <https://ld246.com/article/1562425535552>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

原文链接 [\[每日LeetCode\] 13. Roman to Integer](#)

### Description:

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

Symbol	Value
I	1
V	5
X	10
L	50
C	100
D	500
M	1000

For example, two is written as **II** in Roman numeral, just two one's added together. Twelve is written as, **XII**, which is simply **X**+**II**. The number twenty seven 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 a roman numeral, convert it to an integer. Input is guaranteed to be within the range from 1 to 3999.

### Example 1:

Input: "III"  
Output: 3

### Example 2:

Input: "IV"  
Output: 4

### Example 3:

Input: "IX"  
Output: 9

### Example 4:

Input: "LVIII"  
Output: 58  
Explanation: L = 50, V = 5, III = 3.

### Example 5:

Input: "MCMXCIV"

Output: 1994

Explanation: M = 1000, CM = 900, XC = 90 and IV = 4.

---

思路：本题要求将罗马数字转化为整数。首先理解罗马数字的拼写规则。罗马数字只用来计数，不用演算。罗马数字共有7个，分别为I(1),V(5),X(10),L(50),C(100),D(500),M(1000)。且有以下几个规则

规则一：重复次数

一个罗马数字重复几次，就表示这个数的几倍。最多只能连续出现三次。

规则二：右加左减

在较大的罗马数字的右边记上较小的罗马数字，表示大数字加上小数字。

在较大的罗马数字的左边记上较小的罗马数字，表示大数字减去小数字。

左减的数字必须为一位，比如8写成VIII，而不是IIX。

规则三：加线乘千（本题不考虑）

在罗马数字的上方加一条横线或者加上下标M，表示将这个数乘以1000，加两条横线表示乘以1000平方。

实现转化时只需判断当前数字和下一个数字的大小即可。

---

C++代码

```
class Solution {
public:
    int intval(char c){
        switch(c){
            case 'I': return 1;
                break;
            case 'V': return 5;
                break;
            case 'X': return 10;
                break;
            case 'L': return 50;
                break;
            case 'C': return 100;
                break;
            case 'D': return 500;
                break;
            case 'M': return 1000;
                break;
        }
    }
    return 0;
};
```

```
}  
int romanToInt(string s) {  
    int sum = 0;  
    for(int i = 0; i < s.length(); i++){  
        if(intval(s[i]) < intval(s[i+1])){  
            sum -= intval(s[i]);  
        }  
        else  
            sum += intval(s[i]);  
    }  
    return sum;  
}  
};
```

---

运行时间: 4ms

运行内存: 8.3M