```
class Solution {
    public int romanToInt(String s) {
        int sum = 0, len = s.length();
        int[] arr = new int[len];

        for (int i = 0; i < len; i++) {
            switch(s.charAt(i)) {
                case 'M':
                    arr[i] = 1000;
                    break;
                case 'D':
                    arr[i] = 500;
                    break;
                case 'C':
                    arr[i] = 100;
                    break;
                case 'L':
                    arr[i] = 50;
                    break;
                case 'X':
                    arr[i] = 10;
                    break;
                case 'V':
                    arr[i] = 5;
                    break;
                case 'I':
                    arr[i] = 1;
                    break;
            }
        }

        for (int i = 0; i < len-1; i++) {
            if (arr[i] < arr[i+1]) {
                sum -= arr[i];
            } else {
                sum += arr[i];
            }
        }
        return sum + arr[len-1];
    }
}
```

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 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.
```