

2.Medium\03.Roman_to_Integer.cpp

```
1  /*
2  Question:
3  Given a Roman numeral as a string, convert it to an integer.
4
5  Example 1:
6
7  Input: s = "III"
8  Output: 3
9  Explanation: III = 3.
10 Example 2:
11
12 Input: s = "LVIII"
13 Output: 58
14 Explanation: L = 50, V= 5, III = 3.
15
16
17 Approach:
18 1. Create a map to store the values of each Roman symbol.
19 2. Initialize `result` as the value of the first symbol in the input string.
20 3. Iterate through each character `c` in the input string `s`, starting from the second
   character.
21     a. If the value of the current symbol is greater than the value of the previous symbol,
   subtract twice the value of the previous symbol from `result` and add the value of the
   current symbol.
22     b. Otherwise, add the value of the current symbol to `result`.
23 4. Return `result` as the converted integer.
24
25 Time Complexity: O(n), where n is the length of the input string `s`.
26 Space Complexity: O(1)
27 */
28
29 int romanToInt(string s) {
30     unordered_map<char, int> symbolValues = {
31         {'I', 1},
32         {'V', 5},
33         {'X', 10},
34         {'L', 50},
35         {'C', 100},
36         {'D', 500},
37         {'M', 1000}
38     };
39
40     int result = symbolValues[s[0]];
41     for (int i = 1; i < s.size(); i++) {
42         if (symbolValues[s[i]] > symbolValues[s[i - 1]]) {
43             result -= 2 * symbolValues[s[i - 1]];
44             result += symbolValues[s[i]];
45         } else {
46             result += symbolValues[s[i]];
47         }
48     }
49 }
```

```
50 |         return result;  
51 |     }  
52 |
```