

Преобразование римского числа в целое. Римские цифры представлены семью различными символами: I, V, X, L, C, D и M.

Символ Значение

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

Например, число 2 записывается как II римскими цифрами, просто две сложенные вместе. Число 12 записывается как XII, то есть просто X + II. Число 27 записывается как XXVII, то есть XX + V + II.

Римские цифры обычно пишутся слева направо от большего к меньшему. Однако цифра четыре не IIII. Вместо этого цифра четыре пишется как IV.

Так как единица предшествует пятерке, мы вычитаем ее и получаем четыре. Тот же принцип применим к числу девять, которое пишется как IX.

Есть шесть случаев, когда используется вычитание:

I можно поставить перед V (5) и X (10), чтобы получились 4 и 9.

X можно поставить перед L (50) и C (100), чтобы получилось 40 и 90.

C можно поставить перед D (500) и M (1000), чтобы получились 400 и 900.

Пример 1

Input: "III"

Output: 3

Пример 2

Input: "LVIII"

Output: 58

Пример 3

Input: "MCMXCIV"

Output: 1994

Код программы:

```
main.cs
1 using System;
2 using System.Collections.Generic;
3
4 public class RomanToInteger
5 {
6     public int RomanToInt(string s)
7     {
8         // Создаем словарь, связывающий римские цифры с их числовыми значениями
9         Dictionary<char, int> map = new Dictionary<char, int>()
10         {
11             {'I', 1},
12             {'V', 5},
13             {'X', 10},
14             {'L', 50},
15             {'C', 100},
16             {'D', 500},
17             {'M', 1000}
18         };
19
20         // обработка римского числа с последнего символа
21         int result = map[s.Length - 1];
22
23         // в обратном порядке по остальным римским цифрам
24         for (int i = s.Length - 2; i >= 0; i--)
25         {
26             // Если текущая цифра меньше следующей, вычит
27             if (map[s[i]] < map[s[i + 1]])
28             {
29                 result -= map[s[i]];
30             }
31             // прибавляем ее к результату
32             else
33             {
34                 result += map[s[i]];
35             }
36         }
37
38         // Возвращаем полученный результат
39         return result;
40     }
41
42     public static void Main(string[] args)
43     {
44         RomanToInteger rti = new RomanToInteger();
```

```

13         {'X', 10},
14         {'L', 50},
15         {'C', 100},
16         {'D', 500},
17         {'M', 1000}
18     };
19
20     // обработка римского числа с последнего символа
21     int result = map[s.Length - 1];
22
23     // в обратном порядке по остальным римским цифрам
24     for (int i = s.Length - 2; i >= 0; i--)
25     {
26         // Если текущая цифра меньше следующей, вычитаем
27         if (map[s[i]] < map[s[i + 1]])
28         {
29             result -= map[s[i]];
30         }
31         // прибавляем ее к результату
32         else
33         {
34             result += map[s[i]];
35         }
36     }
37
38     // Возвращаем результат
39     return result;
40 }
41
42 public static void Main(string[] args)
43 {
44     RomanToInteger rti = new RomanToInteger();
45     // 3
46     Console.WriteLine(rti.RomanToInt("III"));
47     // 58
48     Console.WriteLine(rti.RomanToInt("LVIII"));
49     // 1994
50     Console.WriteLine(rti.RomanToInt("MCMXCIV"));
51 }

```

Вывод программы:

```

3
58
1994

...Program finished with exit code 0
Press ENTER to exit console.

```