

Roman to Integer

题目：Given a roman numeral, convert it to an integer.

(眨眼初看题目并没有发现题目到底要我们求的是什么，不知道罗马数字和我们的阿拉伯数字的转换规则到底是怎样的，于是有了如下这张表)

I	1		XXXII	32		LXIII	63		XCIV	94
II	2		XXXIII	33		LXIV	64		XCV	95
III	3		XXXIV	34		LXV	65		XCVI	96
IV	4		XXXV	35		LXVI	66		XCVII	97
V	5		XXXVI	36		LXVII	67		XCVIII	98
VI	6		XXXVII	37		LXVIII	68		XCIX	99
VII	7		XXXVIII	38		LXIX	69		C	100
VIII	8		XXXIX	39		LXX	70			
IX	9		XL	40		LXXI	71		EX.	
X	10		XLI	41		LXXII	72		DI	501
XI	11		XLII	42		LXXIII	73		DL	550
XII	12		XLIII	43		LXXIV	74		DXXX	530
XIII	13		XLIV	44		LXXV	75		DCCVII	707
XIV	14		XLV	45		LXXVI	76		DCCCXC	890
XV	15		XLVI	46		LXXVII	77		MD	1500
XVI	16		XLVII	47		LXXVIII	78		MDCCC	1800
XVII	17		XLVIII	48		LXXIX	79		CM	900
XVIII	18		XLIX	49		LXXX	80			
XIX	19		L	50		LXXXI	81			
XX	20		LI	51		LXXXII	82			
XXI	21		LII	52		LXXXIII	83			
XXII	22		LIII	53		LXXXIV	84			
XXIII	23		LIV	54		LXXXV	85			
XXIV	24		LV	55		LXXXVI	86			
XXV	25		LVI	56		LXXXVII	87			
XXVI	26		LVII	57		LXXXVIII	88			

以上这张表就是对应的转换规则，我们可以看到有几个特殊的字符代表的某个整数，分别是1，5，10，50，100，500，1000.因为题目只要求表示到3999，所以用到1000就足够了。

实现思路：由于涉及到字符还有数字的对应关系，应该想到使用map最好。可以发现做这些题需要经常使用

到map，所以掌握好map的相关语法非常重要。

另外我们需要注意到里面的某些规则，比如4是IV而不是IIII，9又是IX，14是XIV等等这一类的存在前一个字符比后一个字符小的情况。其实这样的情况也只会出现在表示带有4，9这样的情况。所以我们的思路就是从后往前遍历这个字符串，并且对相邻的字符进行判断，假如靠左的字符比靠右的字符要小，那么就减去靠左的字符代表的数字的值，反之就加上该值。

```
int romanToInt(string s) {
    map T = {
        {'I',1},{'V',5},{'X',10},{'L',50},{'C',100},{'D',500},{'M',1000}
    };
    int sum = T[s.back()];
    for (int i = s.length()-2; i >=0 ; i--) {
        if(T[s[i]] < T[s[i+1]]){
            sum -= T[s[i]];
        }
        else{
            sum += T[s[i]];
        }
    }
    return sum;
}
```