Roman numeral to integer

To convert a Roman numeral string s into an integer, here's the key idea:

Core logic:
- Roman numerals are usually additive: "XII" = 10+1+1 = 12
- But when a smaller value appears before a larger one, it's subtractive: "IV" = 5 - 1 = 4

Strategy:
1. Map each symbol to its integer value.
2. Iterate left to right:
   - For each character, look ahead one character.
   - If the current value is less than the next one, subtract it.
   - Otherwise, add it.

Example "MCMXCIV"

Char	Value	Next Char	Next Value	Action	Result
M	1000	C	100	add	1000
C	100	M	1000	subtract	1000-100=900
M	1000	X	10	add	1900
X	10	C	100	subtract	1900-10=1890
C	100	I	1	add	1990
I	1	V	5	subtract	1990-1=1989
V	5	—	-	add	1994

Helper Map: use hash map like:

```
unordered_map<char, int> roman = {{'I', 1}, {'V', 5}, {'X', 10}, {'L', 50},
                                  {'C', 100}, {'D', 500}, {'M', 1000}};
```

Pseudo code:

```
int total = 0;
for each index i from 0 to s.length() - 1:
    value = roman[s[i]]
```

```
if i < s.length() - 1 and value < roman[s[i+1]]:
    total -= value
else:
    total += value
```