First Non Repeating Character in C++

```
#include <iostream>
#include <string>
#include <unordered_map>

using namespace std;

int sol(string s) {
    unordered_map<char, int> fmap;

    // Build frequency map
    for (char c : s) {
        fmap[c]++;
    }

    // Find first non-repeating character
    for (int i = 0; i < s.length(); i++) {
        char ch = s[i];
        if (fmap[ch] == 1) {
            return i;
        }
    }

    return -1; // If no non-repeating character found
}

int main() {
    string s = "abbcaddecfab";
    cout << sol(s) << endl;
    return 0;
}
```

Input:

s = "abbcaddecfab"

Step 1 - Build Frequency Map:

The frequency map (fmap) will look like this:

- 'a' → 2
- 'b' → 3
- 'c' → 2
- 'd' → 2
- 'e' → 2
- 'f' → 1

Step 2 - Find First Non-Repeating Character:

We now iterate through the string and check the frequency of each character:

1. For index 0: s[0] = 'a' → frequency of 'a' is 2 (repeated).
2. For index 1: s[1] = 'b' → frequency of 'b' is 3 (repeated).
3. For index 2: s[2] = 'b' → frequency of 'b' is 3 (repeated).
4. For index 3: s[3] = 'c' → frequency of 'c' is 2 (repeated).
5. For index 4: s[4] = 'a' → frequency of 'a' is 2 (repeated).
6. For index 5: s[5] = 'd' → frequency of 'd' is 2 (repeated).
7. For index 6: s[6] = 'd' → frequency of 'd' is 2 (repeated).
8. For index 7: s[7] = 'e' → frequency of 'e' is 2 (repeated).
9. For index 8: s[8] = 'c' → frequency of 'c' is 2 (repeated).
10. For index 9: s[9] = 'f' → frequency of 'f' is 1 (non-repeating).

Now, the first non-repeating character is 'f', which appears at index **7**, not index **9**.

Conclusion:

- The first non-repeating character in the string "abbcaddecfab" is 'f', which appears at **index 7**.

Output:
7