**Problem-1**

**Aim:**

Given a string s, find the first non-repeating character in it and return its index. If it does not exist, return -1.

Constraints:

1 <= s.length <= 105

s consists of only lowercase English letters.

1. **Problem Description:**

A string s is given by the user and we have to find the first non-repeating character in the string and return index of that character and also if the character does not exist then return -1.

1. **Algorithm :**

Define vector<pair<char, int>> v.

For loop in range (0 to string size)

Set char c=S[i]

Find in range(v.begin() to v.end())

Create a pair of (char, int)

Check if first meet occur push in pair vector

Else push -1

Exit for loop

Initialize ans = -1

Print pair vector as answer

1. **Source Code for Experiment :**

class Solution {

public:

int firstUniqChar(string s) {

        vector<pair<char, int>> v;

        for(int i = 0; i < s.size(); i++){

            char c = s[i];

            auto it = find\_if(v.begin(), v.end(),

                [&c](const pair<char, int>& p){return p.first == c;});

            if(it == v.end()){

                //first meet

                v.push\_back(make\_pair(c, i));

            }else{

                it->second = -1;

            }

        }

        int ans = -1;

        for(pair<char,int>& p : v){

            if(p.second != -1){

                ans = p.second;

                break;

            }

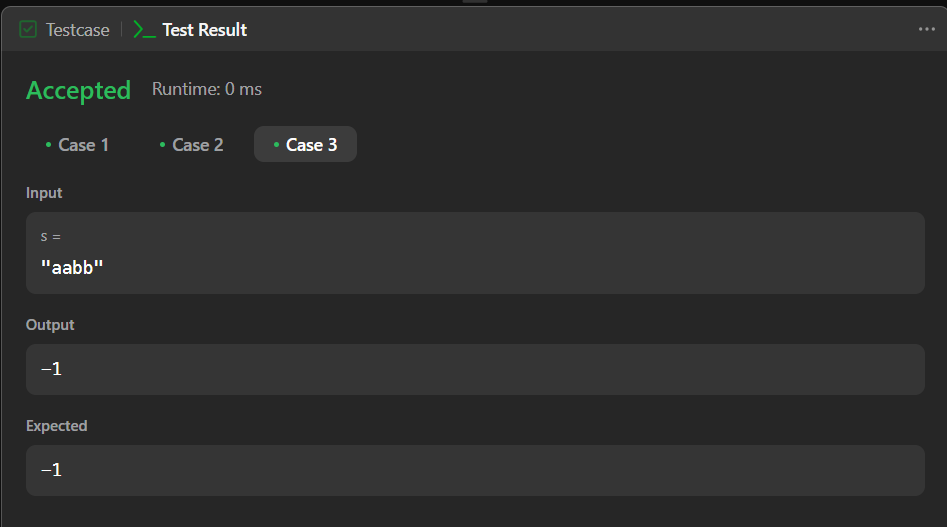
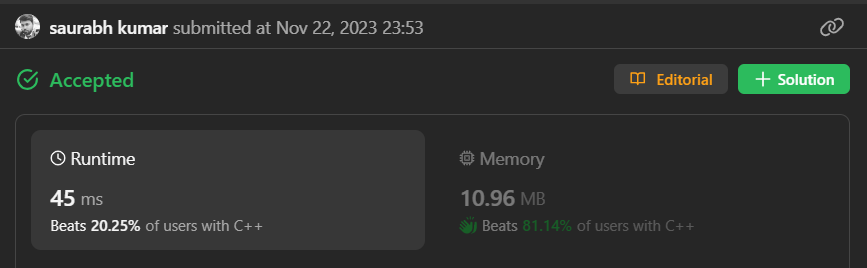
        }

        return ans;

    }

};

1. **Result/Output :**

****

**Learning outcomes (What I have learnt):**

1. I learnt about the string data structure in C++.
2. I learnt about the vector and pair data structure.
3. I learnt about the for loop and if condition statement.
4. I learnt about how to find first unique character in a string.