Experiment-1.2

Student Name: Aayush Gurung UID:20BCS5323

Branch: CSE Section/Group:DM_607(A)

Semester: 6 Date of Performance:24-02-2023

Subject Name: CC LAB Subject Code: 20CST-355

Aim:

To demonstrate the concept of String Matching algorithms

Objective

Problem 1:

Rotate-String

Approach:

Checking if the concatenation of s with itself contains goal

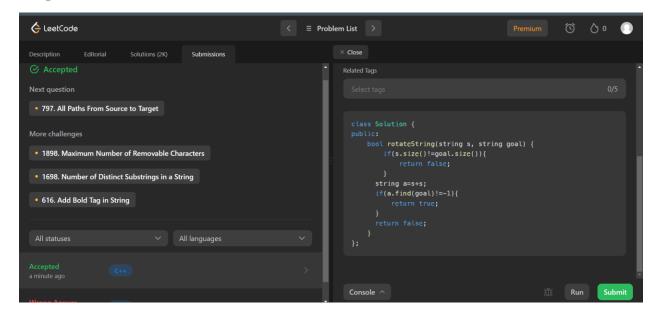
Algorithm:

- 1. Declare a and a=s+s;
- 2. First check if the size of s and goal is same, if not then return false otherwise move to next step
- 3. Then check if a contains goal or not if not then return false else return true.

Code:

```
class Solution {
public:
    bool rotateString(string s, string goal) {
        if(s.size()!=goal.size()){
            return false;
        }
        string a=s+s;
        if(a.find(goal)!=-1){
            return true;
        }
        return false;
    }
}
```

Output:



Problem 2:

Repeated String Match

Approach:

Add the string into a separate variable until it equals given string if its not equal then return -1;

Algorithm:

- 1. Declare the int variables to store the size of a and b string
- 2. Divide both the int variables and add 2 to it to get it equal to the no of times we need to run the loop
- 3. Then run a loop and constantly concatenate the string a to temp variable and check if b is present or not.
- 4. If b is present then return count, if not present then return -1;

Code:

```
class Solution {
public:
    int repeatedStringMatch(string a, string b) {
        int la=a.size();
        int b=b.size();
        int k=(ceil(lb/la))+2;
        int count=0;
        string temp="";
        while(k--){
            temp+=a;
            count++;
            // cout<<temp<<endl;
            if(temp.find(b)!=-1){
                return count;
            }
        }
        return -1;
    }
}</pre>
```

Output:

