TALLER PROBLEMAS BÚSQUEDA Y ORDENAMIENTO

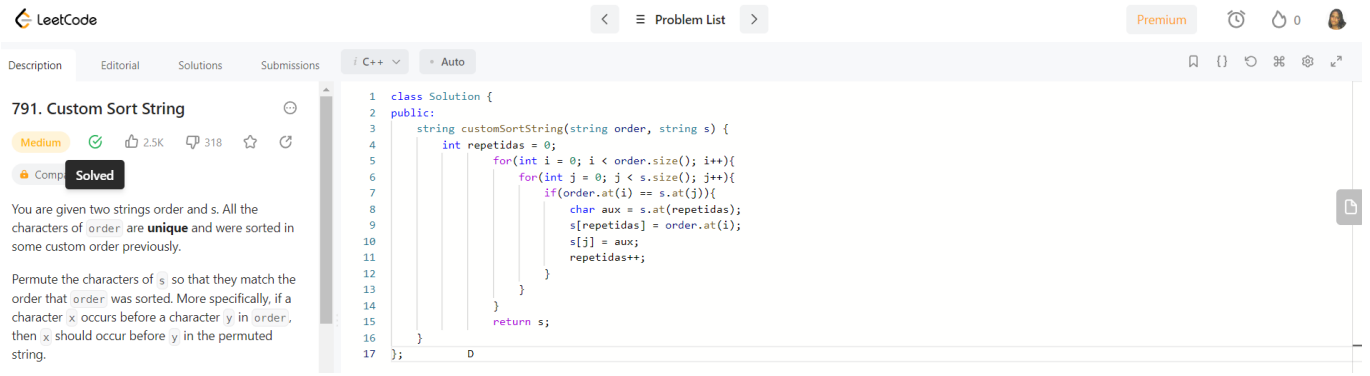
MÓNICA GIL GONZÁLEZ
1152183

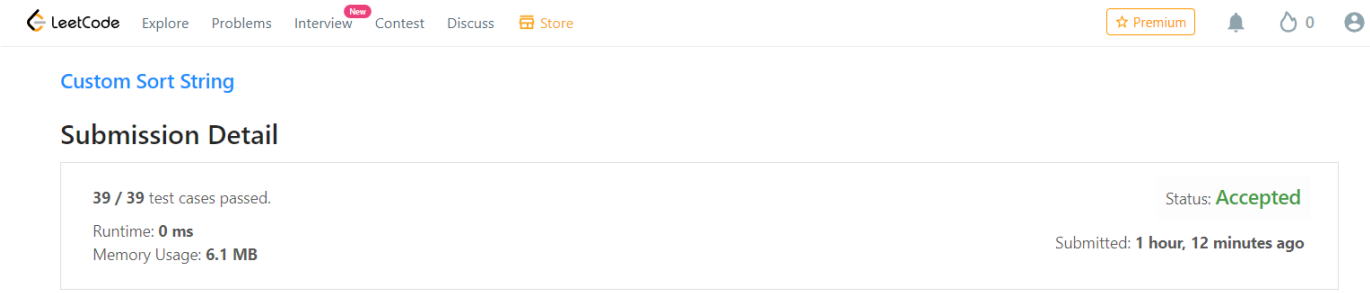
MILTON JESÚS VERA CONTRERAS

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER
INGENIERÍA DE SISTEMAS
ANÁLISIS DE ALGORITMOS
NORTE DE SANTANDER
CÚCUTA
ABRIL 29
2023

1) Resolver el problema en la plataforma https://leetcode.com/

El ejercicio que de manera aleatoria me correspondió en la actividad es “Custom Sort String” adjunto link del mismo y evidencia del ACCEPTED en la plataforma: https://leetcode.com/problems/custom-sort-string/





Adjunto el código:

```
class Solution {
public:
    string customSortString(string order, string s) {
        int repetidas = 0;
                for(int i = 0; i < order.size(); i++){
                    for(int j = 0; j < s.size(); j++){
                        if(order.at(i) == s.at(j)){
                            char aux = s.at(repetidas);
                            s[repetidas] = order.at(i);
                            s[j] = aux;
                            repetidas++;
                        }
                    }
                }
                return s;
    }
};
```

2) Desarrollar una aplicación que genere al menos 100 casos de prueba para el problema. Los casos deben cubrir todas las posibilidades de casos del problema, de manera equilibrada.

En una hoja de Excel se plantearon 100 casos de prueba para el problema analizado en esta actividad:
https://docs.google.com/spreadsheets/d/1KcE4VvHipDEUO1t8kxTKAYs0QBYe9lHxZDfbvWhZZCI/edit?usp=sharing

Para cubrir todas las posibilidades de casos del problema de manera equilibrada establecimos los casos de la siguiente manera:

El string buscado se encuentra en la mitad	10
El string buscado se encuentra de primero	10
El string buscado se encuentra de último	10
El size ORDER es mayor que el size S	10
El size ORDER es menor que el size S	10
El size ORDER es igual que el size S	10
El string buscado no se encuentra completo	10
El string buscado se encuentra completo	10
Algunos caracteres del string buscado cuenta con recurrencias	10
Casos borde	10
TOTAL DE CASOS =	100

Restricciones para los datos de entrada:

- 1 <= ORDER.length <= 26
- 1 <= S.length <= 200
- ODER y S consisten en letras minúsculas.
- Todos los caracteres de ORDER son únicos .

3) Desarrollar una aplicación completa, con main y lectura de datos, de manera que se pruebe la solución independientemente de la plataforma https://leetcode.com/. **Esta solución debe ser consistente con la solución realizada en el numeral 1.**

```
//Monikilla
#include <bits/stdc++.h>
using namespace std;
int main() {
  ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
  string order, s;
  while(getline(cin >> ws, order)){
    getline(cin, s);
    int repetidas = 0;
    for(int i = 0; i < order.size(); i++){
      for(int j = 0; j < s.size(); j++){
        if(order.at(i) == s.at(j)){
          char aux = s.at(repetidas);
          s[repetidas] = order.at(i);
          s[j] = aux;
          repetidas++;
        }
      }
    }
    cout << s << endl;
  }
  return 0;
}
```

4) Elaborar un video explicando en detalle los tres numerales anteriores. Preferiblemente subirlo a Youtube.

Adjunto link al video correspondiente, subido en mi YouTube académico: https://youtu.be/AAR8QjcYVyk

5) Publicar en github todo lo anterior y escribir un informe en PDF que se sube a UVIRTUAL con los links a todo lo anterior.

Link al GitHub: https://github.com/MonicaGilgon/TallerBusquedaYOrdenamiento