Myo Aung

Test Cases & Display

HashMap-UserName-Password | (Global Scope) | main()

```cpp
/**
 * Project-Password-Username Implementation using Hash Map
 * Testing
 * Password.cpp
 * Copyright (c) 2022, Myo Aung.
 */

#include <fstream>
#include <string>
#include <vector>
#include "SeparateChaining.h"
#include "LinearProbing.h"

 using namespace std;

int main() {

    string pw, name, answer;
    HashMap<string, string>* LPHM = new HashMap<string, string>;
    Hash SCHM;
    cout << "------------------------------" << endl;
    cout << "READING FROM \"password.txt\" " << endl;
    cout << "------------------------------" << endl;

    fstream outputFile("password.txt", ios::in);
    string first, second;
    if (outputFile.is_open()) {
        while (!outputFile.eof()) {
        outputFile >> first >> second;
        cout << "["<< first << ", " << second << "]" << endl;

        SCHM.put(first, second);                            // For separate chaining

        LPHM->insert(first, second);                        // For linear probing
        }
    }
    else {
        cout << "ERROR! FILE NOT FOUND!!" << endl;
        return 0;
    }

    LPHM->print();                                          // TESTING LINEAR PROBING
    cout << "The size of the map: " << LPHM->sizeofMap() << endl;
    do {
    cout << "Username: ";
    cin >> name;
    cout << "Password: ";
    cin >> pw;
    LPHM->verify(name, pw);
    cout << "Would you like to continue? " << endl << "Answer(Yes / No): ";
    cin >> answer;
    cout << endl;
    } while (answer == "Yes" || answer == "yes" || answer == "y");
```

```
    SCHM.print();                                          // TESTING SEPARATE CHAINING
    cout << "The size of the map: " << SCHM.size() << endl;
    do{
    cout << "Enter User Name: ";
    cin >> name;
    cout << "Enter Password: ";
    cin >> pw;
    SCHM.verify(name, pw);
    cout << "Would you like to continue? " << endl << "Answer(Yes / No): ";
    cin >> answer;
    cout << endl;
    } while (answer == "Yes" || answer == "yes" || answer == "y");

    return 0;
}
```

Microsoft Visual Studio Debug Console

```
------------------------------
READING FROM "password.txt"
------------------------------
[mario, mushroom_kingdom]
[sonic, gold-rings]
[ellie, no-more-infected]
[snake, cardboard.box]
[chell, cakeisalie]
[pacman, power!pellet]
[lara, tombs4me]
[myo, iloveCS]
[coolkid, killing-it]
[NATE, cool-stuff]
[zaxx, war&peace]
[Avater, 123456]
[dexter, sureThing]
[swooosh, @#$%!]
[T, forArraySeven!]
------------------------------
-- Linear Probing Method--
------------------------------
[0][myo, iloveCS]
[1][sonic, gold-rings]
[2][snake, cardboard.box]
[3][chell, cakeisalie]
[4][coolkid, killing-it]
[5][dexter, sureThing]
[6][ellie, no-more-infected]
[7][Avater, 123456]
[8][mario, mushroom_kingdom]
[9][pacman, power!pellet]
[10][lara, tombs4me]
[11][NATE, cool-stuff]
[12][zaxx, war&peace]
[13][swooosh, @#$%!]
[14][T, forArraySeven!]

--------
The size of the map: 15
Username: sonic
Password: gold-rings
Authorization successful!

Would you like to continue?
Answer(Yes / No): No
```

```
-----------------------------
-- Separate Chaining Method--
-----------------------------
| 0 | -> [myo, iloveCS]
| 1 | -> [sonic, gold-rings]
| 2 | -> [snake, cardboard.box]
| 3 | -> [chell, cakeisalie] -> [dexter, sureThing]
| 4 | -> [coolkid, killing-it]
| 5 | -> [swooosh, @#$%!]
| 6 | -> [ellie, no-more-infected] -> [Avater, 123456]
| 7 | -> [T, forArraySeven!]
| 8 | -> [mario, mushroom_kingdom] -> [pacman, power!pellet] -> [zaxx, war&peace]
| 9 | -> [lara, tombs4me]
|10 | -> [NATE, cool-stuff]

--------
The size of the map: 15
Enter User Name: dexter
Enter Password: sureThing
Authorization successful!

Would you like to continue?
Answer(Yes / No): no
```