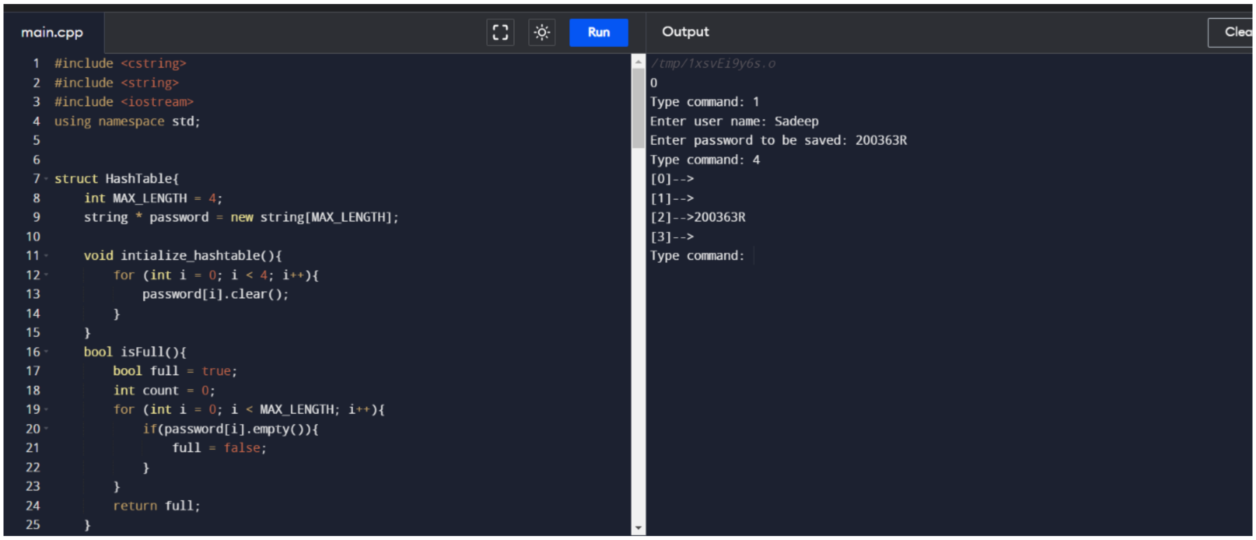
Section 01 :

1)
2)



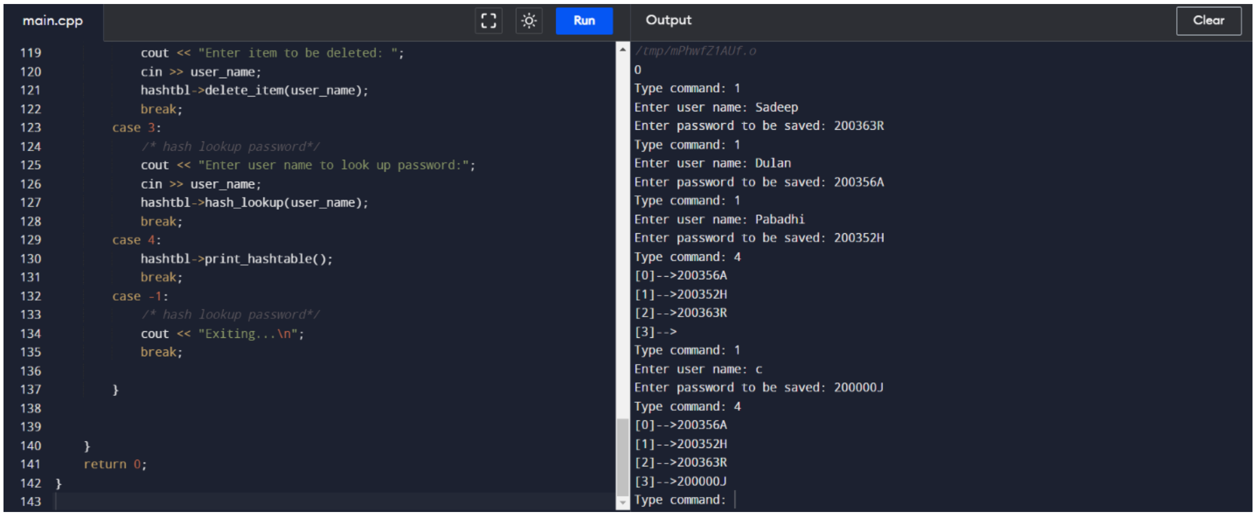
```
#include <cstring>
#include <string>
#include <iostream>
using namespace std;


struct HashTable{
    int MAX_LENGTH = 4;
    string * password = new string[MAX_LENGTH];

    void intialize_hashtable(){
        for (int i = 0; i < 4; i++){
            password[i].clear();
        }
    }
    bool isFull(){
        bool full = true;
        int count = 0;
        for (int i = 0; i < MAX_LENGTH; i++){
            if(password[i].empty()){
                full = false;
            }
        }
        return full;
    }
```

```
/tmp/1xsvEi9y6s.o
0
Type command: 1
Enter user name: Sadeep
Enter password to be saved: 200363R
Type command: 4
[0]-->
[1]-->
[2]-->200363R
[3]-->
Type command:
```

3)



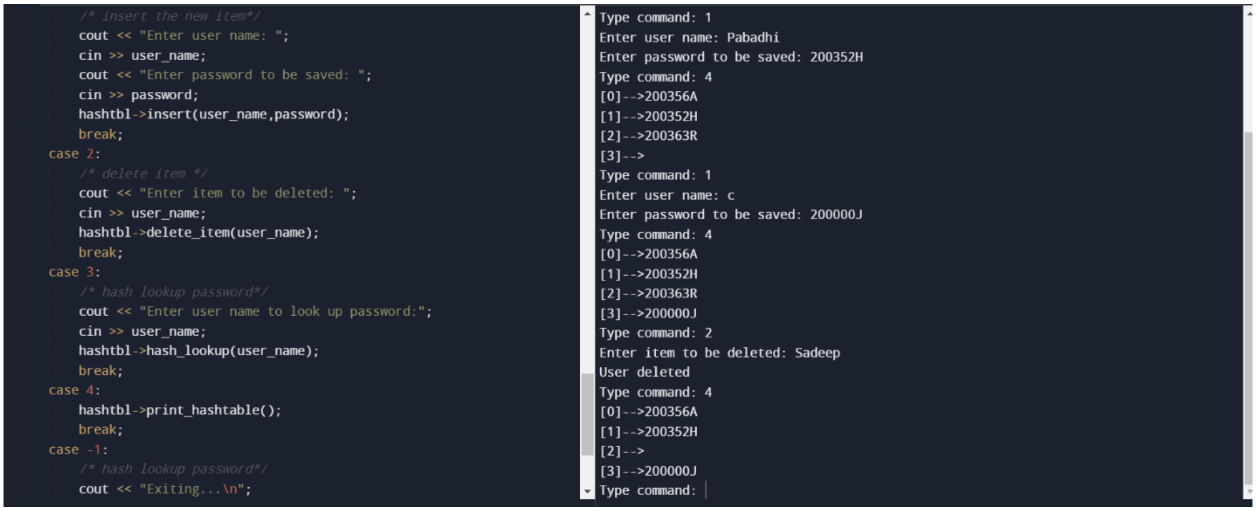
```
            cout << "Enter item to be deleted: ";
            cin >> user_name;
            hashtbl->delete_item(user_name);
            break;
        case 3:
            /* hash lookup password*/
            cout << "Enter user name to look up password:";
            cin >> user_name;
            hashtbl->hash_lookup(user_name);
            break;
        case 4:
            hashtbl->print_hashtable();
            break;
        case -1:
            /* hash lookup password*/
            cout << "Exiting...\n";
            break;

        }


    }
    return 0;
}
```

```
/tmp/mPhwfZ1AUf.o
0
Type command: 1
Enter user name: Sadeep
Enter password to be saved: 200363R
Type command: 1
Enter user name: Dulan
Enter password to be saved: 200356A
Type command: 1
Enter user name: Pabadhi
Enter password to be saved: 200352H
Type command: 4
[0]-->200356A
[1]-->200352H
[2]-->200363R
[3]-->
Type command: 1
Enter user name: c
Enter password to be saved: 200000J
Type command: 4
[0]-->200356A
[1]-->200352H
[2]-->200363R
[3]-->200000J
Type command:
```

4)



```
        /* insert the new item*/
        cout << "Enter user name: ";
        cin >> user_name;
        cout << "Enter password to be saved: ";
        cin >> password;
        hashtbl->insert(user_name,password);
        break;
    case 2:
        /* delete item */
        cout << "Enter item to be deleted: ";
        cin >> user_name;
        hashtbl->delete_item(user_name);
        break;
    case 3:
        /* hash lookup password*/
        cout << "Enter user name to look up password:";
        cin >> user_name;
        hashtbl->hash_lookup(user_name);
        break;
    case 4:
        hashtbl->print_hashtable();
        break;
    case -1:
        /* hash lookup password*/
        cout << "Exiting...\n";
```

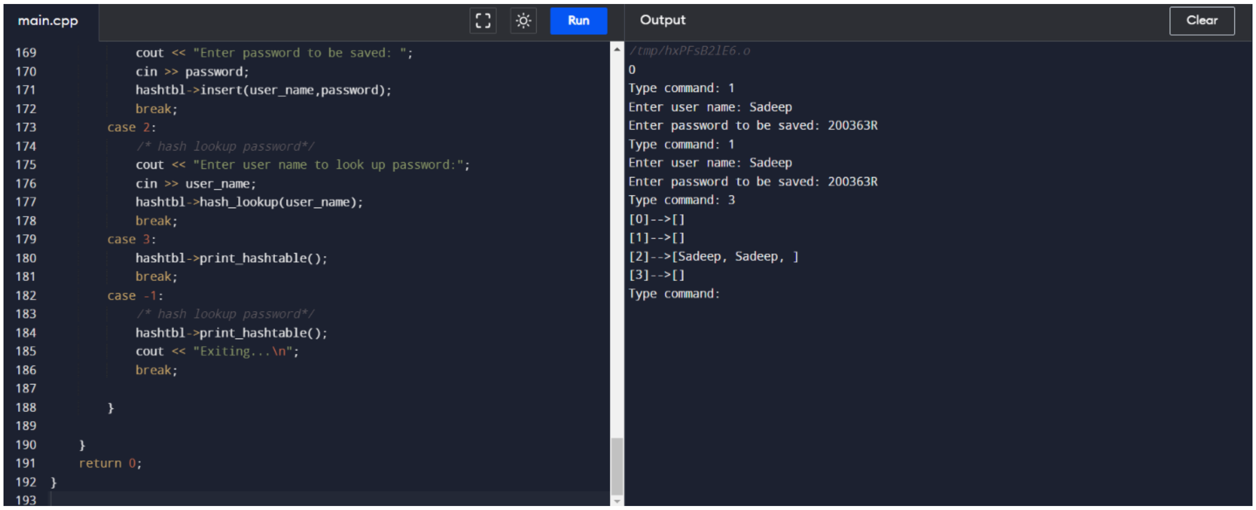
```
Type command: 1
Enter user name: Pabadhi
Enter password to be saved: 200352H
Type command: 4
[0]-->200356A
[1]-->200352H
[2]-->200363R
[3]-->
Type command: 1
Enter user name: c
Enter password to be saved: 200000J
Type command: 4
[0]-->200356A
[1]-->200352H
[2]-->200363R
[3]-->200000J
Type command: 2
Enter item to be deleted: Sadeep
User deleted
Type command: 4
[0]-->200356A
[1]-->200352H
[2]-->
[3]-->200000J
Type command:
```

5) In this table, we can not save multiple passwords in same locations.
Chains with linked lists can be used as a solution for this problem. Then we can save multiple passwords in same location.

Section 2 :

1)

2)



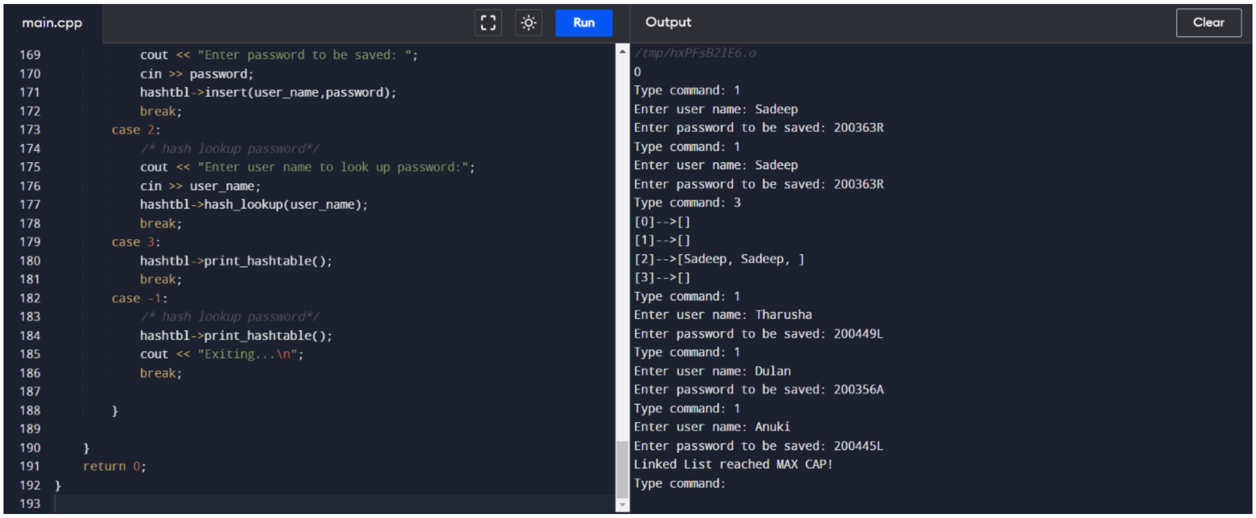
```
            cout << "Enter password to be saved: ";
            cin >> password;
            hashtbl->insert(user_name,password);
            break;
        case 2:
            /* hash lookup password*/
            cout << "Enter user name to look up password:";
            cin >> user_name;
            hashtbl->hash_lookup(user_name);
            break;
        case 3:
            hashtbl->print_hashtable();
            break;
        case -1:
            /* hash lookup password*/
            hashtbl->print_hashtable();
            cout << "Exiting...\n";
            break;

        }

    }
    return 0;
}
```

```
/tmp/hxPFsB21E6.o
0
Type command: 1
Enter user name: Sadeep
Enter password to be saved: 200363R
Type command: 1
Enter user name: Sadeep
Enter password to be saved: 200363R
Type command: 3
[0]-->[]
[1]-->[]
[2]-->[Sadeep, Sadeep, ]
[3]-->[]
Type command:
```

3)



```cpp
            cout << "Enter password to be saved: ";
            cin >> password;
            hashtbl->insert(user_name,password);
            break;
        case 2:
            /* hash lookup password*/
            cout << "Enter user name to look up password:";
            cin >> user_name;
            hashtbl->hash_lookup(user_name);
            break;
        case 3:
            hashtbl->print_hashtable();
            break;
        case -1:
            /* hash lookup password*/
            hashtbl->print_hashtable();
            cout << "Exiting...\n";
            break;

        }

    }
    return 0;
}
```

```
/tmp/hxPFsB21E6.o
0
Type command: 1
Enter user name: Sadeep
Enter password to be saved: 200363R
Type command: 1
Enter user name: Sadeep
Enter password to be saved: 200363R
Type command: 3
[0]-->[]
[1]-->[]
[2]-->[Sadeep, Sadeep, ]
[3]-->[]
Type command: 1
Enter user name: Tharusha
Enter password to be saved: 200449L
Type command: 1
Enter user name: Dulan
Enter password to be saved: 200356A
Type command: 1
Enter user name: Anuki
Enter password to be saved: 200445L
Linked List reached MAX CAP!
Type command:
```

Github link : https://github.com/SadeepRathnayaka/DSA-Labs/tree/main