

Index No. : 210417X

In-Class Lab Exercise – Week 09

Git-Hub link :

<https://github.com/ChethmiNayanathara/Data-Structures-and-Algorithms>

Implementing basic hash table

Annex 1 –

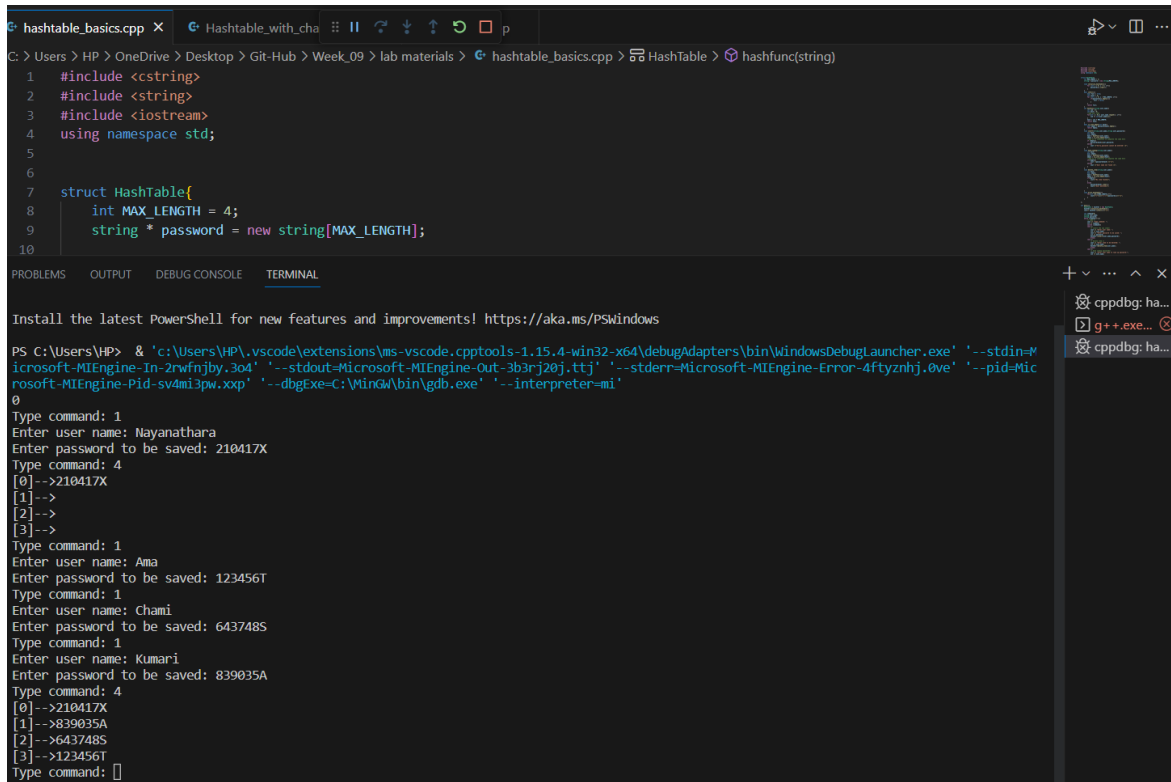
```
hashable_basics.cpp x Hashable_with_cha HashTable > hashfunc(string)
115 cout << "Enter user name to look up password:";
116 cin >> user_name;
117 hashtable->hash_lookup(user_name);
118 break;
119 case 4:
120     hashtable->print_hashtable();
121     break;
122 case -1:
123     /* hash lookup password*/
124     cout << "Exiting...\n";

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\HP> & 'c:\Users\HP\.vscode\extensions\ms-vscode.cpptools-1.15.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=M
icrosoft-MIEngine-In-ic1230rs.omm' '--stdout=Microsoft-MIEngine-Out-ixwd45gj.42u' '--stderr=Microsoft-MIEngine-Error-ojbbdrgrf.vve' '--pid=Mic
rosoft-MIEngine-Pid-vqqprc52.5fb' '--dbgExe=C:\Windows\bin\gdb.exe' '--interpreter=mi'
0
Type command: 1
Enter user name: Nayanathara
Enter password to be saved: 210417X
Type command: 4
[0]-->210417X
[1]-->
[2]-->
[3]-->
Type command: 
```

Annex 2 -

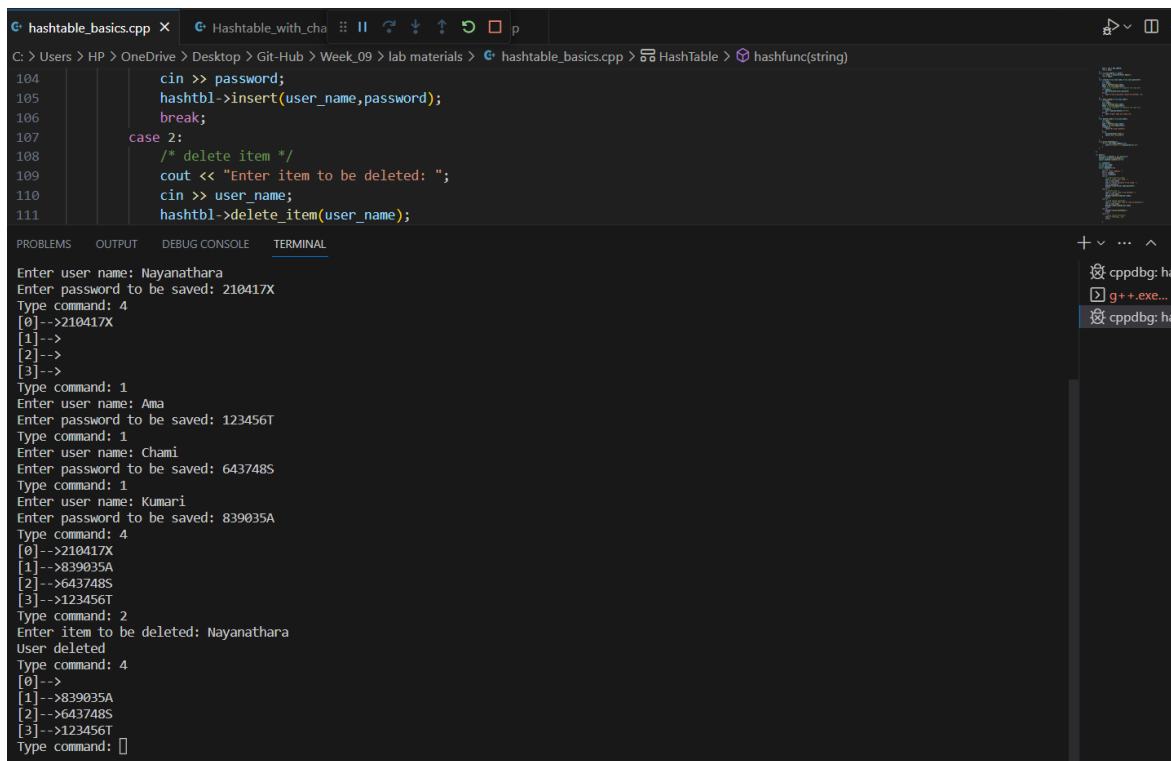


The screenshot shows a Visual Studio Code editor with a C++ file named `hashtable_basics.cpp`. The code defines a `HashTable` struct with a `MAX_LENGTH` of 4 and a `password` array. The `hashfunc` function is implemented to calculate a hash for a string. The terminal output shows the program running and inserting four entries: Nayanathara (210417X), Ama (123456T), Chami (643748S), and Kumari (839035A).

```
1 #include <cstring>
2 #include <string>
3 #include <iostream>
4 using namespace std;
5
6
7 struct HashTable{
8     int MAX_LENGTH = 4;
9     string * password = new string[MAX_LENGTH];
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
PS C:\Users\HP> & 'c:\Users\HP\.vscode\extensions\ms-vscode.cpptools-1.15.4-win32-x64\debugAdapters\bin\windowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-2rwfnjby.3o4' '--stdout=Microsoft-MIEngine-Out-3b3rj20j.ttj' '--stderr=Microsoft-MIEngine-Error-4ftyznjhj.0ve' '--pid=Microsoft-MIEngine-Pid-sv4mi3pw.xop' '--dbgExe=C:\MinGW\bin\gdb.exe' '--interpreter=mi'
0
Type command: 1
Enter user name: Nayanathara
Enter password to be saved: 210417X
Type command: 4
[0]-->210417X
[1]-->
[2]-->
[3]-->
Type command: 1
Enter user name: Ama
Enter password to be saved: 123456T
Type command: 1
Enter user name: Chami
Enter password to be saved: 643748S
Type command: 1
Enter user name: Kumari
Enter password to be saved: 839035A
Type command: 4
[0]-->210417X
[1]-->839035A
[2]-->643748S
[3]-->123456T
Type command: []
```

Annex 3 –



The screenshot shows the same Visual Studio Code editor with the `hashtable_basics.cpp` file. The code now includes a `delete` case in the `hashfunc` function. The terminal output shows the program running and deleting the entry for Nayanathara.

```
104 cin >> password;
105 hashtable->insert(user_name,password);
106 break;
107 case 2:
108     /* delete item */
109     cout << "Enter item to be deleted: ";
110     cin >> user_name;
111     hashtable->delete_item(user_name);
```

```
Enter user name: Nayanathara
Enter password to be saved: 210417X
Type command: 4
[0]-->210417X
[1]-->
[2]-->
[3]-->
Type command: 1
Enter user name: Ama
Enter password to be saved: 123456T
Type command: 1
Enter user name: Chami
Enter password to be saved: 643748S
Type command: 1
Enter user name: Kumari
Enter password to be saved: 839035A
Type command: 4
[0]-->210417X
[1]-->839035A
[2]-->643748S
[3]-->123456T
Type command: 2
Enter item to be deleted: Nayanathara
User deleted
Type command: 4
[0]-->
[1]-->839035A
[2]-->643748S
[3]-->123456T
Type command: []
```

When implementing hash tables in this manner, various keys can result in the same hash value and thereby collisions happen. And also this results in very long linked lists. Hence it becomes inefficient in searching and extracting data.

To overcome these difficulties techniques such as chaining and open addressing can be used.

1. Chaining

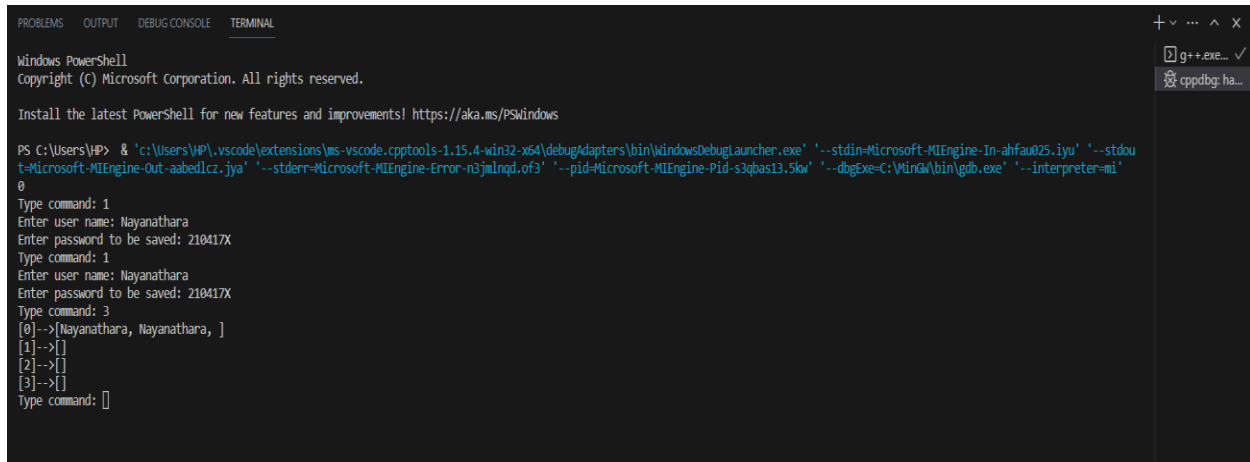
- In this technique, each slot in the hash table is a pointer to a linked list of key-value pairs.
- When collisions occur, the new key-value pairs are simply added to the linked list of the corresponding slot.
- By this method, a large number of collisions can be handled without increasing the size of the hash table.

2. Open addressing

- All the colliding keys are stored in a different location in the hash table itself when the natural choice is full.
- Searching for this different location is known as probing.
- The exact open addressing strategy depends on the implementation of the probing method. There are various probing methods such as linear probing, quadratic probing and doubling hashing.

Implementing hash table with chaining

Annex 4 -



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\HP> & 'c:\Users\HP\.vscode\extensions\ms-vscode.cpptools-1.15.4-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin-Microsoft-MIEngine-In-ahfau025.iyu' '--stdout-Microsoft-MIEngine-Out-aabelcz.jya' '--stderr-Microsoft-MIEngine-Error-n3jmlnqd.of3' '--pid-Microsoft-MIEngine-Pid-s3qbas13.5kw' '--dbgExe=C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.29.30133\bin\amd64\x64\cl.exe' '--interpreter-mi'
Type command: 1
Enter user name: Nayanathara
Enter password to be saved: 210417X
Type command: 1
Enter user name: Nayanathara
Enter password to be saved: 210417X
Type command: 3
[0]-->[Nayanathara, Nayanathara, ]
[1]-->[]
[2]-->[]
[3]-->[]
Type command: []
```

Annex 5 -



```
Enter password to be saved: 210417X
Type command: 1
Enter user name: Nayanathara
Enter password to be saved: 210417X
Type command: 3
[0]-->[Nayanathara, Nayanathara, ]
[1]-->[]
[2]-->[]
[3]-->[]
Type command: 1
Enter user name: Waruni
Enter password to be saved: 234748E
Type command: 1
Enter user name: Anura
Enter password to be saved: 123456A
Type command: 1
Enter user name: Menuri
Enter password to be saved: 234568X
Linked List reached MAX CAP!
Type command: 3
[0]-->[Nayanathara, Nayanathara, ]
[1]-->[]
[2]-->[Waruni, ]
[3]-->[Anura, ]
Type command: []
```