DSA Lab – 1 / 12 / 2021

Double Hashing

Submitted By: Vishal Teotia (19BME1133)

Link:
https://gist.github.com/vashuteotia123/4d6edd3993380df283ea78ee72b68d5d

Code:

```
#include <bits/stdc++.h>

using namespace std;

const int SIZE = 10;



class DoubleHashing {

private:

int *arr;

int current_size;

map<int, int> conflicts;

map<int, int> probes;



public:

DoubleHashing() {

arr = new int[SIZE];

current_size = 0;

for (int i = 0; i < SIZE; i++) {

arr[i] = -1;

}

}

bool isFull() {

if (current_size == SIZE) {
```

```
return 1;
} else {
return 0;
}
}
int Hash1(int key) {
return ((2 * key + 3));
}
int Hash2(int key) {
return ((3 * key + 1));
}
void insert(int key) {
if (isFull()) {
cout << "Sorry! Hash Table is full." << endl;
cout << "You can't insert more values." << endl;
} else {
int index = Hash1(key);
if (arr[index] != -1) {
int index2 = Hash2(key);
int i = 1, cnt = 0, conflict_index;
bool conflict = true;
while (cnt < 100) {
probes[key]++;
int newindex = (index + i * index2) % SIZE;
conflict_index = newindex;
if (arr[newindex] == -1) {
arr[newindex] = key;
conflict = false;
break;
```

```
}
i++;
cnt++;
}
if (conflict) {
conflicts[key] = conflict_index;
probes.erase(key);
}
} else {
arr[index] = key;
}
}
current_size++;
}
void display() {
cout << "Content of Hash Table: " << endl;
for (int i = 0; i < SIZE; i++) {
if (arr[i] != -1) {
cout << i << ": " << arr[i] << endl;
} else {
cout << i << ":" << endl;
}
}
}


void show_conflicts() {
cout << "Element cannot be inserted into hash table." << endl;
for (auto i : conflicts) {
```

```
cout << " Key " << i.first << " conflicted at index " << i.second << endl;
}
cout << endl;
}

void show_probes() {
cout << "Number of probes required for each key. " << endl;
for (auto i : probes) {
cout << " Key " << i.first << " took " << i.second << " iterations" << endl;
}
cout << endl;
}
};

int main() {
cout <<
"\033[1;31m================================================================================
===============\033[0m" << endl;
cout << "\033[1;31m| Topic: Double Hashing - DSA LAB WORK 01-DEC-2021 - Submitted by: Vishal Teotia
(19BME1133) |\033[0m" << endl;
cout <<
"\033[1;31m================================================================================
===============\033[0m" << endl;

int a[] = {3, 2, 9, 6, 11, 13, 7, 12, 15, 5};
int s = sizeof(a) / sizeof(a[0]);
DoubleHashing h;
for (int i = 0; i < s; i++) {
h.insert(a[i]);
```

```
}

h.display();

h.show_conflicts();

h.show_probes();

}
```

Output:

```
SA/LAB/1 DEC/"double-hashing
=============================================================================================
| Topic: Double Hashing - DSA LAB WORK 01-DEC-2021 - Submitted by: Vishal Teotia (19BME1133) |
=============================================================================================
Content of Hash Table:
0:
1: 7
2:
3: 11
4: 6
5: 9
6:
7: 2
8: 12
9: 3
Element cannot be inserted into hash table.
   Key 5 conflicted at index 3
   Key 13 conflicted at index 9
   Key 15 conflicted at index 3

Number of probes required for each key.
   Key 6 took 1 iterations
   Key 7 took 2 iterations
   Key 9 took 3 iterations
   Key 11 took 2 iterations
   Key 12 took 3 iterations
```