Optional Project – 7

Team : Karthika, Nevhetha, Kritika

---

3. Seperate Chaining and Two choice hashing - (Open Hashing)

For a input of random 100000 numbers of the range 0...100000

Separate Chaining yields below results:

```
Max length of the list ::1045
Number of empty lists ::0
```

Two choice hashing yields below results:

```
Max length of the list ::160
Number of empty lists ::14
```

This indicated that linear probing of separate chaining keeps adding to same linked list so almost all the lists are filled and there are no empty list
Whereas two choice hashing with two hash functions adds in varied linked list based on the sizes, so there are empty lists which are not filled and maximum length is lesser than separate chaining lists length.

---

4. Comparison of JAVA hashing with Closed hashing algorithms

For a input of random 100000 numbers of the range 0...100000

```
DoubleHashing:
Time: 478 msec.
Memory: 52 MB / 150 MB.
--------------------------------------
HopScotchHashing:
Time: 500 msec.
Memory: 53 MB / 150 MB.
--------------------------------------
Hashing:
```

Time: 715 msec.
Memory: 53 MB / 204 MB.

From the above results, it is evident that Double hashing and Hop Scotch hashing techniques of Closed hashing is better than the normal JAVA hashing.

----

5. Compare the performances of Trees, Lists, Skip lists, and Hashing on the operations add, contains, remove.

	Time for 100 elements (msec)	Time for Millions of elements (msec)	Time for 10Millions of elements (msec)
Hashing	7	469	2578
Lists	7	4295	> 3 mins
Trees	9	4100	> 2 mins
SkipList	10	756	5488

Hashing behaves faster than Skip List which in turn is better than Trees and Lists
The difference is negligible for small inputs like 100s, but illuminate for Millions and 10 Millions of inputs.