David Ivanov

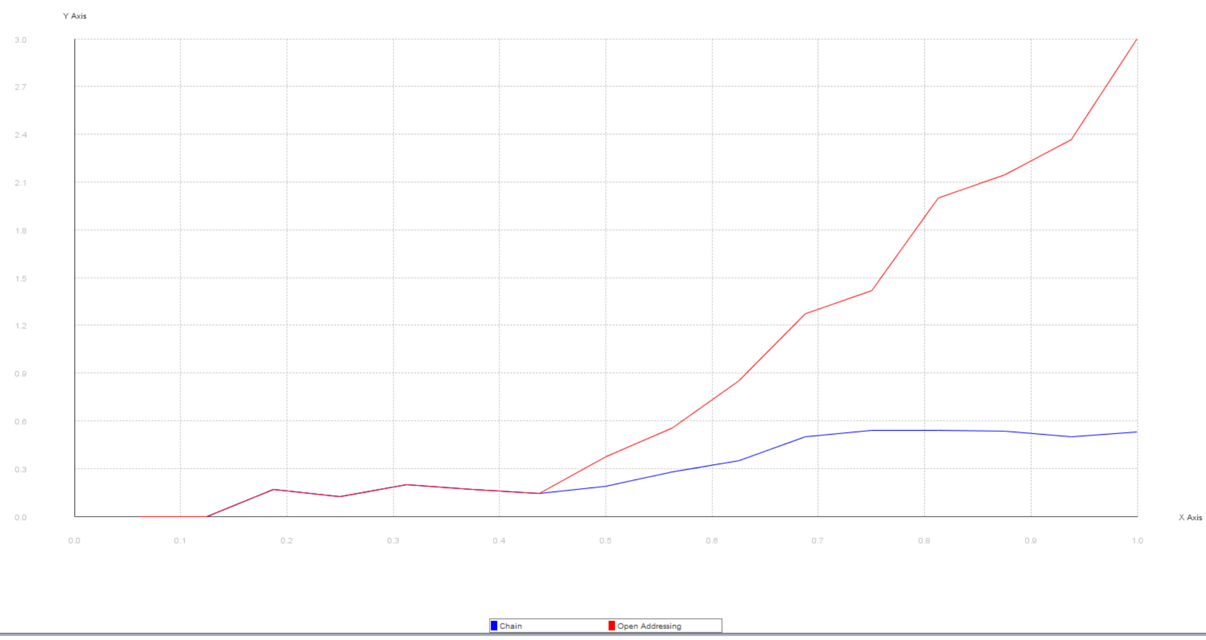
October 9th, 2018

COMP 251

ID: 260806837

## Conclusions

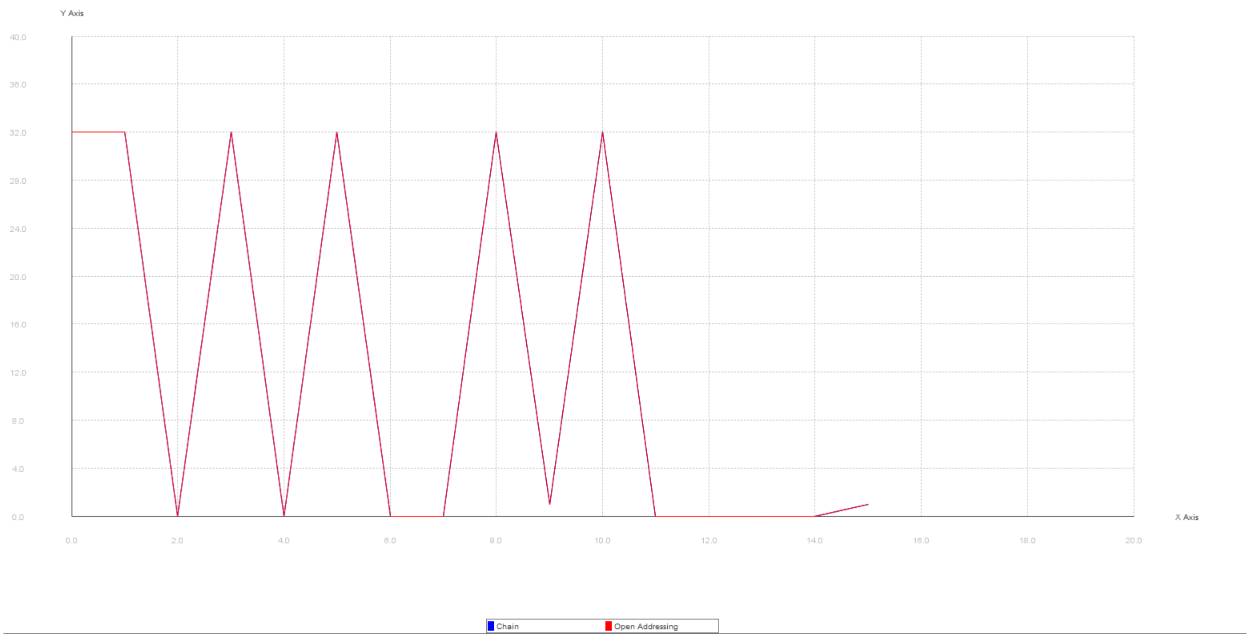
Task 1:




By the graph, a trend appears in both the chaining method and open addressing method for hash tables. Above, the x-axis is the load factor and the y-axis is the average number of collisions per insert. This is experiment was built in a way so that number of inputs were increased to vary the load factor and making a visible effect on the number of collisions. For the open addressing, the hash table tends to fill up relatively fast. The rise of the number of keys increases the likely hood of a collision as keys with the same hash value have higher chance of happening. When a collision happens, the key ends up taking the slot of another key and "technically stealing" its slot. Due to this, the collisions become more frequent as spots which are forced to be taken by the probing of the hash value are blocked and often force the next key to collide upon its insertion. This is the cause of the parabolic trend observed.

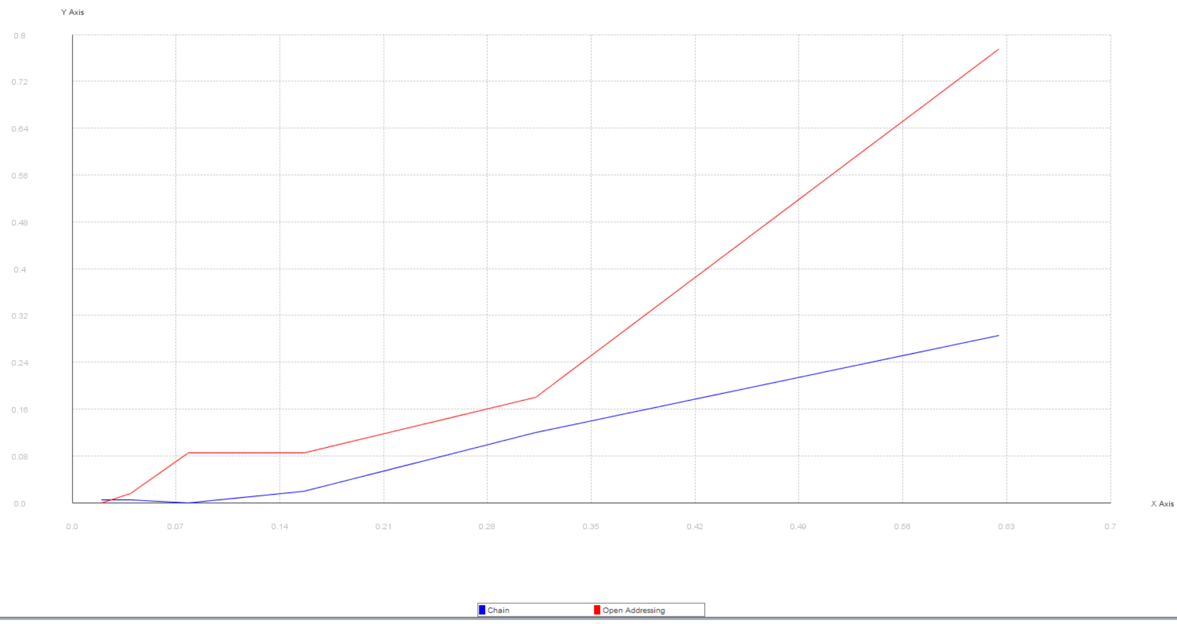
For the chaining method for hash tables, collisions behave differently. For example, if we keep inserting the same key the collisions increase linearly. This is the trend we seem to see in the chaining method. As the number of inputs increase so will the collisions but a certain key can only collide with values in the same hash slot. Therefore, the average collisions appear to form a linear trend and due to this behavior, the collision chance is lower than that of the open addressing.

Task 2:




The y-axis is the collisions during the removal processes and the x-axis is the index of the removal array. What is observed in the graph is that if a key is exists in the hash table there is between 0 and 31 collisions depending on how many encounters the probe forces before it reaches the key it is looking for which explains the zig zag in the graph. If the key cannot be found in the hash table, the returned value is the number of slots which in this case is 32. The number of collisions should decrease as more of the keys are removed as they are more likely to be in the place of the first probe of the hash function would have placed them.

Task 3:




In this experiment, the value of w was varied for each simulation. The y-axis is the average collisions and the x-axis is the load factor for each simulation. From what we know, the value of m is dependent exponentially to r and r is directly proportional to w. Since the load factor is inversely proportional to m then w is inversely exponentially proportional to the load factor. Therefore, if w increases then the load factor should decrease drastically. So as w increases the number of slots increases as well making the chances of collisions in open-addressing and chaining less likely as shown in the graph. As the load factor increases we also see a rise in the number of collisions similarly to what was observed in task 1.

Collaborator: Marco Guida