Hashing Analysis Report

Trial 1:

* Collision Resolution Scheme – Separate Chaining
* Hash Function – Key Mod

In this plot we see generally linear lines with a slight concave up as size increases. The curvature of these line means that the amount of collisions that take place grow roughly proportionally with the load factor. The combination of separate chaining and key mod table size seems to be the best choice theoretically as the fewest amount of collisions take place.

Trial 2:

* Collision Resolution Scheme – Separate Chaining
* Hash Function – Mid Square

In this plot the lines are still generally linear, but this time with a bit more concavity as size increases. The increased concavity in this plot versus the last is due to the use of mid square over key mod table size. This means although separate chaining keeps the lines generally linear, mid square increases the amount of collisions. The combination of separate chaining and mid square is still a relatively good choice as it performed the second best in this experiment.

Trial 3:

* Collision Resolution Scheme – Open Addressing
* Hash Function – Key Mod

In this plot the lines are concaved up, and as the size of the table increases the trend heads towards a more exponential look. The exponentiality of this plot is due to the use of open addressing. As seen in the plot, with smaller tables open addressing does fairly well, but with an increase in load factor and table size comes a drastic increase in the amount of collisions that take place. The combination of separate chaining and key mod table size is something that might benefit smaller tables over larger ones as it performed the third best in this experiment.

Trial 4:

* Collision Resolution Scheme – Open Addressing
* Hash Function – Mid Square

In this plot the again the lines are concaved up, and as the size of the table increases the trend heads towards a more exponential look. The difference is in this table the exponentiality is greater than that of the last. This is because of the use of mid square over key mod table size. Like in the separate chaining plots, the use of the mid square hash functions seems to increase the amount of collisions that take place. The combination of open addressing and mid square performed the worst in terms of collisions.

Collision Resolution Scheme on collisions vs load factor:

* When analyzing the data, it is obvious that the collision resolution scheme has the biggest impact on collisions vs load factor. As seen in the trials above separate chaining had significantly better results when it came to collisions. Open-addressing is usually faster than separate chaining when the load factor is low because you don't have to follow pointers between list nodes, but as the load factor approaches 1 it becomes much slower, and this problem is emphasized as the table size grows larger. The downside of separate chaining is that you must follow pointers in order to search through the linked list at each table index, but the benefit is that the separate chaining only gets linearly slower as the load factor approaches 1.

Hash Function on collisions vs load factor:

* When compared to the effect the collision resolution scheme, the hash function plays less of a role on the collisions vs load factor, but it still has its differences. When looking at the data you can see the key mod table size performs better that mid-square does. This observation is true in both collision resolutions schemes as well.