Project03_Analysis

Nate Hydukovich, Ava De La Cruz, Ryan Vo

October 2025

Introduction

In this experiment, we compared two different hash functions for storing integer keys in a hash table:

1. Most Significant Bits (MSB) Method: uses the upper p bits of the key as the hash value.
2. Cormen's Multiplication Method: multiplies each key by a constant $A = (\sqrt{5} - 1)/2$, takes the fractional part, and scales it by the table size m.

We wanted to see which of these two methods spreads keys more evenly across the hash table.

Experimental Setup

We used data from the file logins.csv, which contains a list of numeric identifiers similar to login records. Each key was inserted into a hash table with $m = 1024$ slots. For each hash function, we counted the number of collisions, the largest bucket size, and how balanced the buckets were.

Results

Hash Function	**Collisions**	**Max Bucket**	**Std. Dev.**
Most Significant Bits (MSB)	3	2	0.31
Multiplication (Cormen/Knuth)	6	2	0.32

Discussion

The MSB method produced fewer collisions and nearly the same standard deviation as the multiplication method. This indicates that, for the data in logins.csv, the MSB approach distributed the keys slightly more evenly across the table. This likely happened because the numeric identifiers in the dataset have more variation in their higher bits, which the MSB method uses directly. The multiplication method performed similarly overall, but did not provide a noticeable improvement in distribution for this dataset.

Conclusion

Based on the experiment, the Most Significant Bits (MSB) method performs slightly better for our data. It resulted in fewer collisions and comparable balance across the table. However, in general cases or with more sequential or structured data, Cormen's multiplication method may still be the safer and more consistent choice.