

# Project03\_Analysis

Nathan Hydukovich

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)	318	27	6.41
Multiplication (Cormen/Knuth)	202	19	4.22

## Discussion

The MSB method produced more collisions and a higher standard deviation. This happened because many of the keys in `logins.csv` share similar high bits, which means the MSB method places many of them in the same buckets. The multiplication method, on the other hand, mixes all bits of the key, spreading the data more evenly.

## Conclusion

Based on the experiment, Cormen's multiplication method performs better. It gives a more uniform distribution of keys, fewer collisions, and better overall balance in the hash table. For most practical uses, especially when data is structured or sequential, this method is the safer choice.