Analysis

Shaina and Annie

March 2023

To compare the performance of the two hash functions, we will generate a large number of keys, hash them using both methods, and then compare the number of collisions.

For this experiment, we will use m = 2^16 and p = 16 for the Most Significant Bits Method. We will generate a list of random 32-bit keys and hash them using both methods we will count the number of collisions in each table using the following code:

```
// Count collisions for Most Significant Bits Method
    int collisions1 = 0;
    for (auto bucket : table1) {
        int n = 0;
        for (Node* node = bucket; node != nullptr; node = node->next) {
            n++;
        }
        collisions1 += n * (n - 1) / 2;
        while (bucket != nullptr) {
            Node* temp = bucket;
            bucket = bucket -> next;
            delete temp;
        }
          }

// Count collisions for Cormen's Multiplication Method
    int collisions2 = 0;
    for (auto bucket : table2) {
        int n = 0;
        for (Node* node = bucket; node != nullptr; node = node->next) {
            n++;
        }
        collisions2 += n * (n - 1) / 2;
        while (bucket != nullptr) {
            Node* temp = bucket;
            bucket = bucket->next;
            delete temp;
        }
    }

    cout << "Most Significant Bits Method: " << collisions1 << " collisions" << endl;
    cout << "Cormen's Multiplication Method: " << collisions2 << " collisions" << endl;

    return 0;
```

```
Most Significant Bits Method: 48987303 collisions
Cormen's Multiplication Method: 1204008 collisions
```

The results show that Cormen's Multiplication Method has significantly fewer collisions than the Most Significant Bits Method. This is expected since the Cormen's Multiplication Method is designed to spread keys more evenly across the hash table. In contrast, the Most Significant Bits Method only uses a small part of the key to determine the hash value, which can result in more collisions. Therefore, we can conclude that Cormen's Multiplication Method is a better hash function for this experiment.