# CS211 – Hash Tables

**Open Addressing – Methods:**

- **Linear Probing:**
  - Move linearly looking for the next available space
    - Start by modding the key value by the hash table size
    - If that slot is full – then move to the next one
    - Repeat until a free slot is found
- **Quadratic Probing:**
  - Systematic jumps to look for a free space, step sizes get bigger and bigger to avoid clustering
    - Start by modding the key value by the hash table size
    - Jump to this slot $x$
    - If this slot is full, then jump to the slot $(x + step^2)$, where step is the step size (1, 2, 3, … etc)
      - Example: First we probe x, then x + 1, then x+4, then x + 9
      - $x, x + 1^2, x + 2^2, x + 3^2, etc.$
    - Do this until a free slot is found
- **Double Hashing:**
  - Using a different hash function to generate a unique jump size for items that generated the same hash index
    - Start by modding the key value by the hash table size – **primary hash function**
    - Then use the secondary hash function to generate the step size
      - The step size is given by: $MAX - (x \% MAX)$ where x is the given unique key

**Load Factor:** The ratio between the size of the hash table and the number of items in it:

- Load Factor of 1 = as many items as there are slots
- Load Factor of 0 = hash table is empty
- With *Open Addressing* the max Load Factor is 1 – each slot can only hold one item
- With separate chaining the load factor can be higher because each slot holds a linked list