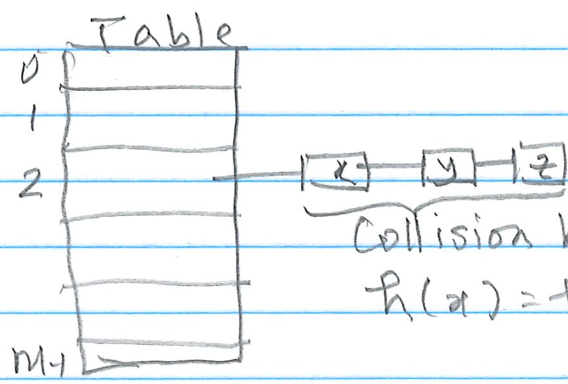
Hashing

- Keys come from some large universe.
- $S \subseteq U$ we actually care about. $|S| = N$. $N \ll |U|$.
- A function $h: U \to \{0, 1, 2, \dots, M-1\}$ where the codomain is the hash table. Generally, $M = O(N)$.
- A collision happens when $h(x) = h(y)$ when $x \neq y$. We will handle collisions with a linked list



Collision keys are chained together
$h(x) = h(y) = h(z) = 2$.

- find, insertion, deletion properties are easy.
- $h(x)$ is easy to implement. Ex $h(x) = B * x \bmod M$ where B is a multiplier
- Hashing is good if the keys of S are uniformly distributed.
- How to counter this? Good hash functions should be random.

Universal Hashing

Def^n: A hashing $h: U \to \{0, 1, \ldots, M-1\}$ is universal, if for all $x \neq y$ in U, we have

$$\Pr(h(x) = h(y)) \leq \frac{1}{M}$$

Constructing a universal hash family

Matrix method $h : U \to \{0, 1, \dots, M-1\}$

- keys are u-bits long
- $M = 2^b$; index is b-bits long.

Consider h to be a random $b \times u$ 0/1 matrix.
$h(x) = hx$ where we do addition mod 2.

$$
\begin{array}{ccccc}
h & & x & & h(x) \\
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} & & \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} & = & \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}
\end{array}
$$

Claim for $x \neq y$, $\Pr_h\left[h(x) = h(y)\right] = \frac{1}{M} = \frac{1}{2^b}$.

Proof. Multiply $\Rightarrow$ add the columns whose corresponding entries in x is 1.

Consider an arbitrary pair of keys x & y, $x \neq y$. Suppose $x_i = 0$ & $y_i = 1$. Now consider all hashing matrix whose i^{th} column is 0. Over the remaining choices of i^{th} column, $h(x)$ is the same. There are 2^b different ways to set the i^{th} column. Each such setting gives a different values of $h(y)$. Therefore, there are $\frac{1}{2^b}$ chance of getting $h(x) = h(y)$.

Creating a hash table for S.

1. Generate a random ~~xxx~~ 0/1 matrix $h_{b\times u}$

2. For any key $x = \langle x_0, x_1, x_2, \ldots, x_{u-1}\rangle$, Compute $hx \bmod 2$. Let $\langle z_0, z_1, \ldots, z_{b-1}\rangle$ be the index.

3. Store x in location $\text{Table}[z_0 2^0 + z_1 \cdot 2^1 + \cdots + z_{b-1} \cdot 2^{b-1}]$

# of expected Collision : $\frac{N}{2^b} = \frac{N}{M}$.

If $M \in O(N)$, # of expected collisions is constant.

$\therefore$ Insert, Find & Delete operations are $O(1)$.