# Hashing and Dictionaries

→ Ordered dict: ↳ gib less than query

under → member, pred, succ, insert, delete } bst pretty good

↳ if we relax pred and succ we can do better

→ keys = {0, 1, ..., u-1} assume usually $u = 2^{64}$

↳ trivial implementation → $|S| = n$

↳ Allocate array $T[0, \dots, n-1]$

→ member is array search } unique hash per val

→ insert, del trivial

→ but u could bigger than n! Also this is shit ← too big

## Hash + Chain

### Hashing

- $|S| = n$
- Allocate an array $T[0..m-1]$ ($m$ comparable to $n$)
  - $T[i]$ is a linked list.
- Choose a ***hash function*** $h : \{0, \dots, u-1\} \to \{0, \dots, m-1\}$ from some "family" of functions $H$.
- Insert$(x, \text{val})$ : Append $(x, \text{val})$ to $T[h(x)]$ ← $O(1 + |T[h(x)]|)$ time + time to compute $h(x)$
  - $x$ and $y$ ***collide*** if $h(x) = h(y)$
- Delete$(x)$ : Remove $(x, \text{val})$ from $T[h(x)]$
- Member$(x)$ : Return true if $x$ is found in $T[h(x)]$

### What is a good "family" of functions

- The "gold standard" is *h is selected from the set of all functions from* $\{0, \dots, u-1\} \to \{0, \dots, m-1\}$.
  - How many functions of this type are there?

    $m^u$
  - How much space does it take to represent such an $h$?

    $u \log m$ bits

    ↳ u space → ↳ storing in table too big
- This is called the ***IDEAL Hash Model***. Choosing a uniformly random function is completely impractical. We want to find a family that mimics the properties of the Ideal Hash Model.

What is good in a hash func? $h: \{0, \dots, n-1\} \to \{0, \dots, m-1\}$

→ Uniformity → $\forall x, i \quad \Pr_{h \in H}(h(x) = i) = 1/m$

→ Collising → $\forall x \neq y \quad \Pr_{h \in H}(h(x) = h(y)) = 1/m$

→ Dependence → $\forall x_1, \dots, x_k, i_1, \dots, i_k$

↳ k wise

$$\Pr_{h \in H}(h(x_1) = i_1, \dots, h(x_k) = i_k) = \prod_{j=1}^{k} \Pr_{h \in H}(h(x_j) = i_j) = 1/m^k$$

## Prob 1

$\Omega$ → ctbl set of outcomes, $p: \Omega \to \mathbb{R}$ so sum of $\sum_{\omega \in \Omega} p(\omega) = 1$

Event $E_1$ → subset of $\Omega$ → ind if $E_1, E_2$ then $P(E_1 \cap E_2) = P(E_1)P(E_2)$

Random variable is func $X: \Omega \to \mathbb{R}$

Exp → $E(X) = \sum_{s \in \Omega} P(s) X(s) = \sum_{x} x \cdot \Pr(X = x)$

→ random var is ind if $[X=k_1]$, $[Y=k_2]$ ind $\forall k_1, k_2$

→ if X, Y ind $E[X \cdot Y] = E[X]E[Y]$

→ Lin of exp.

→ if F an event. Indicator rv. $I_F = \begin{cases} 0 & \text{if } F \text{ occurs} \\ 1 & \text{if } x \text{ does} \end{cases}$

→ $E[I_F] = P(I_F = 1) = \sum_{x \in F} P(x)$

Note: $1+x \le e^x \Rightarrow \left(1+\frac{1}{n}\right)^n < e < \left(1+\frac{1}{n}\right)^{n+1}$

$\left(1-\frac{1}{n}\right)^n < e < \left(1-\frac{1}{n}\right)^{n-1}$

## Hashing with Chaining

$S = \{y_1, \dots, y_n\}$ member (x)?

time $\geq |T[h(x)]|$

↳ linked list in array T at has value of h(x)

What is $E(|T[h(x)]|) = \sum_{k=0}^{n} k \Pr(|T[h(x)]| = k)$

$I_i \quad 1 \le i \le n = \begin{cases} 1 & \text{if } h(x) = h(y_i) \\ 0 & \text{otherwise} \end{cases}$

$\Pr(h(x) = h(y_i)) = P(I_i = 1) = 1/m$

$E(|T[h(x)]|) = E\left(\sum_{i=1}^{n} I_i\right) = \sum_{i=1}^{n} E(I_i)$

$= \sum_i \Pr[h(x) = h(y_i)]$

$= \begin{cases} \frac{1}{m} & x \notin S \\ 1 + \frac{n-1}{m} & x \in S \end{cases}$

### Construct a Good Hash Function

- Pick a prime number $p > u$.
- Hash family $H = \{h_{a,b} : a \in \{1, \dots, p-1\}, b \in \{0, \dots, p-1\}$.
  - $g_{a,b}(x) = ax + b \bmod p$
  - $h_{a,b}(x) = g_{a,b}(x) \bmod m$
- **Claim.** For all $x \neq y$, and $r, s \in \{0, \dots, m-1\}$,
  - $\Pr(h(x) = h(y)) \le 1/m$.
  - $\Pr(h(x) = r \text{ and } h(y) = s) = \Theta(1/m^2)$.

→ let $x \neq y$ → let choice of $a, b$

$\Pr_{a,b}(g(x) = i \cap g(y) = j)$

$\Pr_{a,b}(ax+b = i \bmod p \cap ay+b = 1 \pmod p)$

$\Rightarrow a(x-y) \equiv i-j \pmod p$ ($a \neq 0$)

$\Rightarrow \begin{cases} a \equiv (i-j)(x-y)^{-1} \to 1/p-1 \\ b \equiv i - ax \to 1/p \end{cases}$

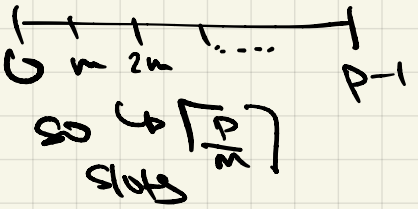
mult inverse mod p prime

$\frac{1}{p(p-1)}$ $\forall i \neq j$

$\Rightarrow \Pr_{a,b}(g(x)=i \wedge g(y)=j) = \begin{cases} 0 & i=j \\ \frac{1}{p(p-1)} & i \neq j \end{cases}$ → diff (int hash to same

$\rightarrow \Pr(h(x)=h(y)) = \frac{|\{(i,j) \mid i=j \pmod m\}|}{p(p-1)}$ → sample

$g(x) \equiv g(y) \bmod m$




so $\leq \lceil \frac{p}{m} \rceil$ slots

$\leq \frac{p\left(\lceil \frac{p}{m} \rceil - 1\right)}{p(p-1)}$

$= \frac{\lceil \frac{p}{m} \rceil - 1}{p-1}$

$\leq \frac{\frac{p-1}{m}}{p-1} \Rightarrow \frac{1}{m}$

$P(h(x)=r \wedge h(y)=s) = \frac{|\{(i,j) \mid i \bmod m = r,\ j \bmod m = s\}|}{p(p-1)}$

$p > 3m$

worst case $\lceil \frac{p}{m} \rceil$ (more for (md)

$\leq \frac{\lceil \frac{p}{m} \rceil \lceil \frac{p}{m} \rceil}{p(p-1)} \in O(\frac{1}{m^2})$

lower bd $\geq \frac{(\lceil \frac{p}{m} \rceil - 1)(\lceil \frac{p}{m} \rceil - 2)}{p(p-1)}$ diff vals

$\in O(1$