

Universal hashing:

- choose a random hash function h from \mathcal{H}
- require \mathcal{H} to be a universal hash family:
$$\Pr_{h \in \mathcal{H}} \{h(k) = h(k')\} \leq \frac{1}{m} \text{ for all } k \neq k'$$
- now just assuming h is random
- no assumption about input keys
(like Randomized Quicksort)

Theorem: for n arbitrary distinct keys
& for random $h \in \mathcal{H}$, & \mathcal{H} universal
 $E[\# \text{ keys colliding in a slot}] \leq 1 + \alpha$
 $\hookrightarrow n/m$

Proof: - consider keys k_1, k_2, \dots, k_n
- let $I_{i,j} = \begin{cases} 1 & \text{if } h(k_i) = h(k_j) \\ 0 & \text{else} \end{cases}$ INDICATOR
RANDOM
VARIABLE

$$\begin{aligned} E[\# \text{ keys hashing to same slot as } k_i] &= E\left[\sum_{j=1}^n I_{i,j}\right] \\ &= \sum_{j=1}^n E[I_{i,j}] \leftarrow \text{linearity of expectation} \\ &= \sum_{j \neq i} E[I_{i,j}] + E[I_{i,i}] \\ &= \Pr\{I_{i,j}=1\} \leftarrow \text{indicator random var.} \\ &= \Pr\{h(k_i) = h(k_j)\} \leftarrow \text{def. of } I_{i,j} \\ &\leq 1/m \leftarrow \text{universality} \\ &\leq n/m + 1 \quad \square \end{aligned}$$

\Rightarrow Insert, Delete, Search cost $O(1 + \alpha)$ expected.

Theorem: dot-product hash family \mathcal{H} is universal

Proof: take any two keys $k \neq k'$

\Rightarrow differ in some digit, say $k_d \neq k'_d$

— let $\text{not } d = \{0, 1, \dots, r-1\} \setminus \{d\}$

$$\begin{aligned} & \Pr_a \{h_a(k) = h_a(k')\} \\ &= \Pr_a \left\{ \sum_{i=0}^{r-1} a_i \cdot k_i = \sum_{i=0}^{r-1} a_i \cdot k'_i \pmod{m} \right\} \\ &= \Pr_a \left\{ \sum_{i \neq d} a_i \cdot k_i + a_d \cdot k_d = \sum_{i \neq d} a_i \cdot k'_i + a_d \cdot k'_d \pmod{m} \right\} \\ &= \Pr_a \left\{ \sum_{i \neq d} a_i (k_i - k'_i) + a_d (k_d - k'_d) = 0 \pmod{m} \right\} \\ &= \Pr_a \left\{ a_d = - \underbrace{(k_d - k'_d)^{-1}}_{\substack{m \text{ prime} \Rightarrow \mathbb{Z}_m \text{ has} \\ \text{multiplicative} \\ \text{inverses}}} \sum_{i \neq d} a_i (k_i - k'_i) \pmod{m} \right\} \\ &= \mathbb{E}_{a_{\text{not } d}} \left[\Pr_{a_d} \{a_d = f(k, k', a_{\text{not } d})\} \right] \leftarrow \begin{array}{l} \text{because } a_d \\ \text{is independent} \\ \text{from } a_{\text{not } d} \end{array} \\ & \quad \left(= \sum_x \Pr\{a_{\text{not } d} = x\} \Pr_{a_d} \{a_d = f(k, k', x)\} \right) \\ &= \mathbb{E}_{a_{\text{not } d}} \left[1/m \right] \\ &= 1/m \end{aligned}$$

□

Another universal hash family: [CLRS]

— choose prime $p \geq u$ (once)

— $h_{ab}(k) = [(a \cdot k + b) \bmod p] \bmod m$

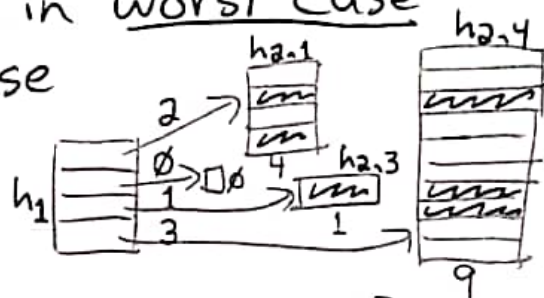
— $\mathcal{H} = \{h_{ab} \mid a, b \in \{0, 1, \dots, u-1\}\}$

Static dictionary problem: given n keys to store in table, support $\text{Search}(k)$.

→ no collisions
Perfect hashing: [Fredman, Komlós, Szemerédi 1984]

- polynomial build time w.h.p. (nearly linear)
- $O(1)$ time for Search in worst case
- $O(n)$ space in worst case

Idea: 2-level hashing



- pick $h_1: \{0, 1, \dots, u-1\} \rightarrow \{0, 1, \dots, m-1\}$ from a universal hash family for $m = \Theta(n)$ (e.g. nearby prime)
 - hash all items with chaining using h_1

- for each slot $j \in \{0, 1, \dots, m-1\}$:
 - let $l_j = \# \text{ items in slot } j = |\{i \mid h(k_i) = j\}|$
 - pick $h_{2,j}: \{0, 1, \dots, u-1\} \rightarrow \{0, 1, \dots, m_j\}$ from a universal hash family for $l_j^2 \leq m_j \leq O(l_j^2)$ (e.g. nearby prime)
 - replace chain in ① slot j with hashing-with-chaining using $h_{2,j}$

$$\text{Space} = O(n + \sum_{j=0}^{m-1} l_j^2)$$

- to guarantee space $= O(n)$:

- if $\sum_{j=0}^{m-1} l_j^2 > cn$ then redo step ①

Search time = $O(1)$ for first table (h_1)
 + $O(\text{max chain size in second table})$
 - to guarantee = $O(1)$:

(2.5) while $h_{2,j}(k_i) = h_{2,j}(k_{i'})$ for any $i \neq i' \wedge j$:
 repick $h_{2,j}$ & rehash those l_j items \rightarrow ANY collision

\Rightarrow no collisions at second level!

Build time: (1) & (2) are $O(n)$. (1.5) & (2.5)?

$$\begin{aligned} (2.5): & \Pr_{h_{2,j}} \{ h_{2,j}(k_i) = h_{2,j}(k_{i'}) \text{ for some } i \neq i' \} \\ & \leq \sum_{i \neq i'} \Pr_{h_{2,j}} \{ h_{2,j}(k_i) = h_{2,j}(k_{i'}) \} \quad \leftarrow \text{Union Bound} \\ & \leq \binom{l_j}{2} \cdot \frac{1}{l_j} \quad \leftarrow \text{by universality} \\ & < \frac{1}{2} \quad (\text{Birthday Paradox}) \end{aligned}$$

\Rightarrow each trial is like a coin flip, tails \Rightarrow OK

$\Rightarrow E[\# \text{ trials}] \leq 2$

& $\# \text{ trials} = O(\lg n)$ w.h.p. (by Lecture 7)

- Chernoff bound $\Rightarrow l_j = O(\lg n)$ w.h.p.

\Rightarrow each trial $O(\lg n)$ time (also obviously $O(n)$)

- must do this for each j

$\Rightarrow O(n \lg^2 n)$ time w.h.p. (or obviously $O(n^2 \lg n)$)

$$\begin{aligned}
 \textcircled{1.5}: E\left[\sum_{j=0}^{m-1} l_j^2\right] &= E\left[\sum_{i=1}^n \sum_{i'=1}^n \underbrace{I_{i,i'}}_{\text{indicator rand. var.} = \begin{cases} 1 & \text{if } h_1(k_i) = h_1(k_{i'}) \\ 0 & \text{else} \end{cases}}\right] \\
 &= \sum_{i=1}^n \sum_{i'=1}^n E[I_{i,i'}] \leftarrow \text{linearity of expectation} \\
 &= \sum_{i=1}^n E[I_{i,i}] + 2 \sum_{i \neq i'} E[I_{i,i'}] \\
 &\leq n + 2 \binom{n}{2} \cdot 1/m \leftarrow \text{universality} \\
 &= O(n) \text{ because } m = \Theta(n) \\
 \Pr_{h_1} \left\{ \sum_{j=0}^{m-1} l_j^2 \geq c \cdot n \right\} &\leq \frac{E\left[\sum_{j=0}^{m-1} l_j^2\right]}{c \cdot n} \quad \left. \vphantom{\Pr_{h_1}} \right\} \text{Markov inequality}
 \end{aligned}$$

$\leq 1/2$ for suff. large const. c
 $\Rightarrow E[\# \text{ trials}] \leq 2$
 $\& \# \text{ trials} = O(\lg n)$ w.h.p.
 $\Rightarrow \textcircled{1} \& \textcircled{1.5}$ take $O(n \lg n)$ w.h.p.