6.046
Lecture 8

Universal Hashing, Perfect Hashing

Weakness of hashing:

For any choice of hash function
$\exists$ a bad set of keys that all hash to same slot.

potential vulnerability against an adversary

Idea: choose hash function at random, independently from keys

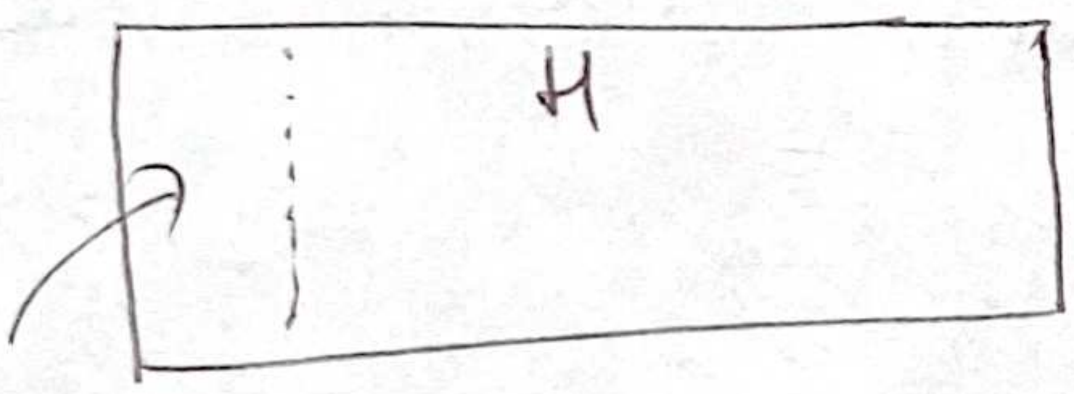
Universal hashing:

Def. Let U be a universe of keys, and let H be a finite collection of hash functions mapping U to $\{0, 1, \ldots, m-1\}$

H is universal if $\forall x, y \in U$, where $x \neq y$,

$$|\{h \in H : h(x) = h(y)\}| = \frac{|H|}{m}$$

i.e. if h is chosen randomly from H, the prob. of collision between x and y is $1/m$



$\underbrace{\{h : h(x) = h(y)\}}_{\frac{1}{m}|H|}$ ← this subset of H would be different for each x, y distinct pair

Thm: Choose h randomly from H. Suppose hashing n keys into m slots in table T. Then, for a given key x,
$E[\#\text{ collisions with } x] < \frac{n}{m}$

Pf. Let C_x be r.v. denoting total # collisions of keys in T with x,

let $c_{xy} = \begin{cases} 1 & \text{if } h(x) = h(y) \\ 0 & \text{otherwise} \end{cases}$

$E[c_{xy}] = 1/m$

$C_x = \sum_{y \in T - \{x\}} c_{xy}$

$$E[C_x] = E\Big[\sum_{y \in T - \{x\}} c_{xy}\Big]$$
$$= \sum_{y \in T - \{x\}} E[c_{xy}] = \sum_{y \in T - \{x\}} \frac{1}{m}$$
$$= \frac{n-1}{m} < \frac{n}{m}$$ ← exp. guarantee against adversary

Constructing a universal hash function

Let m be prime. Decompose key k into $r+1$ digits:

$k = \langle k_0, k_1, \ldots, k_r \rangle$, where $0 \le k_i \le m-1$

↖ Representation of k base m

base: $m, m^2, \ldots, m^{r+1}$ then use mod

- pick $a = \langle a_0, a_1, \ldots a_r \rangle$, each a_i is chosen randomly from $\{0, 1, \ldots m-1\}$
- Define $h_a(k) = \left(\sum_{i=0}^{r} a_i k_i\right) \bmod m$

  ↖ dot product a and k, then take mod [redacted] m

How big is H?

$|H| = m^{r+1}$ ← # of all a

<u>Thm:</u> H is universal

↙ base m representations

Pf. Let $x = \langle x_0, x_1, \ldots x_r \rangle$

$y = \langle y_0, y_1, \ldots y_r \rangle$ be distinct keys

$\Rightarrow$ they differ in at least one digit,

<u>wlog</u> position 0.

For how many $h_a \in H$ do x and y collide?

Must have $h_a(x) = h_a(y)$

$$\Rightarrow \sum_{i=0}^{r} a_i x_i \equiv \sum_{i=0}^{r} a_i y_i \pmod m$$

↑ congruent

$$\Rightarrow \sum_{i=0}^{r} a_i (x_i - y_i) \equiv 0 \pmod m$$

$$\Rightarrow a_0(x_0 - y_0) + \sum_{i=1}^{r} a_i (x_i - y_i) = 0 \bmod (m)$$

$$\Rightarrow a_0(x_0 - y_0) \equiv -\sum_{i=1}^{r} a_i (x_i - y_i) \bmod (m)$$

$2 \equiv -5 \bmod 7$

<u>Number theory fact:</u>

Let <u>m be prime</u>. For any $z \in \mathbb{Z}_m$ (integers mod m)

s.t. $z \neq 0$, $\exists$ unique $z^{-1} \in \mathbb{Z}_m$ s.t. $z \cdot z^{-1} \equiv 1 \pmod m$.

↖ not true if m [redacted] is not prime since any $z \in \mathbb{Z}_m$ is relatively prime to m

<u>Ex</u> $m = 7$

z	1	2	3	4	5	6
z^{-1}	1	4	5	2	3	6

$a \pmod b$

↖ ↗ if not relatively prime a does not have an inverse mod b.

6.046
Lecture 8

Since $x_0 \neq y_0$, $\exists\ (x_0 - y_0)^{-1}$

$$\Rightarrow a_0 \equiv \left(-\sum_{i=0}^{r} a_i (x_i - y_i)\right) \cdot (x_0 - y_0)^{-1} \pmod m$$

- if x, y hash to the same place (assumed) then a_0 has a particular value as a function of other a_i
- Thus, for any choice of $a_1, a_2 \dots a_r$ exactly 1 of m choices of a_0 that causes x and y to collide, and no collision for other m−1 choices for a_0

$$\Rightarrow \#\ h_a\text{'s that cause } x, y \text{ to collide} = \underset{a_1}{m} \cdot \underset{a_2}{m} \cdots \underset{a_r}{m} \cdot \underset{a_0}{1}$$

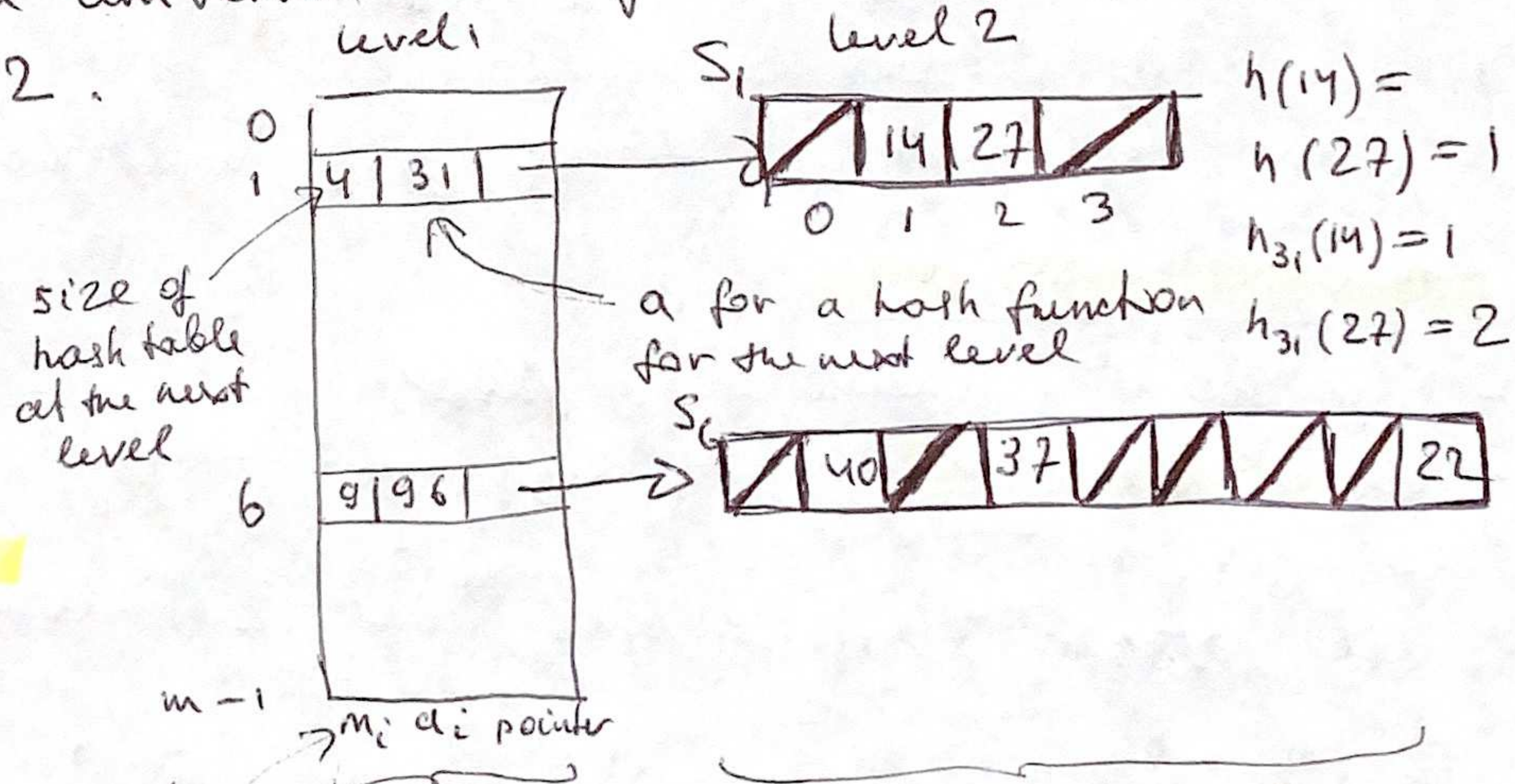
$$= m^r = \frac{|H|}{m} \checkmark$$

Perfect Hashing (given a fixed set of keys, built a static table with good worst case search time) → not in expectation

Given n keys, construct a static hash table of size $m = O(n)$, s.t. search takes $O(1)$ time in the worst case.

Two-level scheme with universal hashing at both levels.
No collisions at level 2.

If n_i items hash to level 1 slot i, then use $m_i = n_i^2$ slots in level 2 table S_i



use another univers. hashing working on non-primes

not prime here, but could pick primes close to these values

universal hashing, pick $h \in H$ at random

universal hashing with an $h \in H$ picked for each slot of level 1, at random

Level 2 analysis:

Thm: Hash n keys into $m = n^2$ slots, using random h in universal $H \Rightarrow E[\#\text{ collisions}] < \frac{1}{2}$

Pf: Prob. 2 given keys collide under h is $\frac{1}{m} = \frac{1}{n^2}$

$\binom{n}{2}$ pairs of keys

$$E[\#\text{ collisions}] = \binom{n}{2}\frac{1}{n^2} = \frac{n(n-1)}{2}\frac{1}{n^2} = \frac{\cancel{n^2}}{2\cancel{n^2}} - \frac{\cancel{n}}{2n^{\cancel{2}}} = \frac{1}{2} - \frac{1}{2n} < \frac{1}{2} \checkmark$$

Markov ineq:

For r.v. $X \geq 0$, $Pr\{X \geq t\} \leq \frac{E(X)}{t}$

Pf. $$E[X] = \sum_{x=0}^{\infty} x\, Pr\{X = x\} \geq \sum_{x=t}^{\infty} x\, Pr\{X = x\}$$

throw away lower terms

$$\geq \sum_{x=t}^{\infty} t\, Pr\{X = x\} = t\, Pr\{X \geq t\} \checkmark$$

Corollary

$Pr\{\text{no collisions}\} \geq \frac{1}{2}$

Pf: $Pr\{\geq 1 \text{ collision}\} \leq \frac{E[\#\text{ collisions}]}{1} < \frac{1}{2} \checkmark$

To find a good level-2 hash function, just test a few at random. Find one quickly, since $\geq 1/2$ will work. } **randomized construction**

[redacted]

check a fixed # of hash functions for each slot, s.t. the prob. to find one without collisions is very high

Analysis of storage

- For level 1, choose $m = n$
- let n_i be r.v. for # keys that hash to slot i in T.

Use n_i^2 slots in each level-2 table S_i.

$$E[\text{total storage}] = n + E\left[\sum_{i=0}^{m-1} \Theta(n_i^2)\right]$$

$$= \Theta(n) \text{ by bucket sort analysis}$$