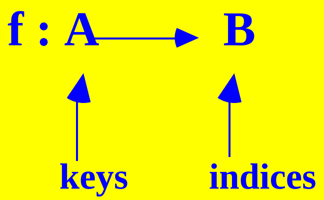
Algorithms
Professor John Reif

ALG 4.2

Universal Hash Functions:

CLR - Chapter 34
Auxillary Reading Selections:
AHU-Data Section 4.7
BB Section 8.4.4
Handout: Carter & Wegman, "Universal Classes of Hash Functions", JCSS, Vol. 18, pp. 143-154, 1979.

Hash Function



f has *conflict* at x,y ε A if
$x \neq y$ but $f(x) = f(y)$

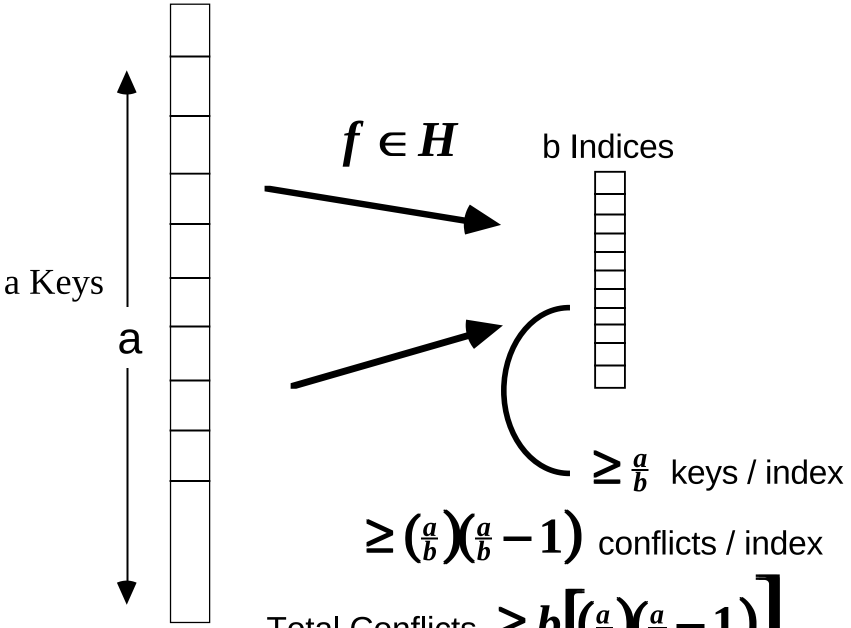
$$\sigma_f(x,y) = \begin{cases} 1 & \text{if } x \neq y \text{ and } f(x) = f(y) \\ 0 & \text{else} \end{cases}$$

If H is a set of hash functions,

$$\sigma_H(x,y) = \sum_{f \varepsilon H} \sigma_f(x,y)$$

for set of keys S,

$$\sigma_H(x,S) = \sum_{f \varepsilon H} \sum_{y \varepsilon S} \sigma_f(x,y)$$



$\geq \frac{a}{b}$ keys / index

$\geq \left(\frac{a}{b}\right)\left(\frac{a}{b}-1\right)$ conflicts / index

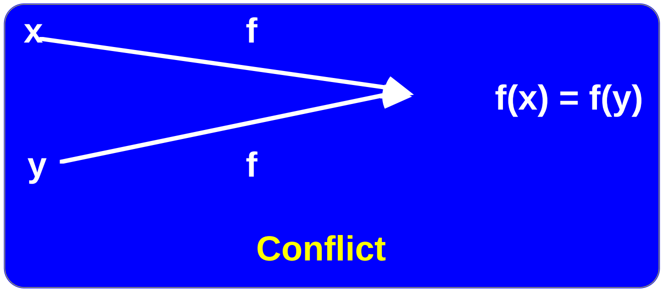
Total Conflicts $\geq b\left[\left(\frac{a}{b}\right)\left(\frac{a}{b}-1\right)\right]$

$\geq \frac{a^2}{b} - a$

H is a *universal* $_2$ set of hash functions

if $\sigma_H(x,y) \le \frac{|H|}{|B|}$ **for all x, y ε A**

i.e. no pair of keys x,y are mapped into the same index by $> \frac{1}{|B|}$ of all functions in H



Proposition 1

Given *any* set H of hash fn,

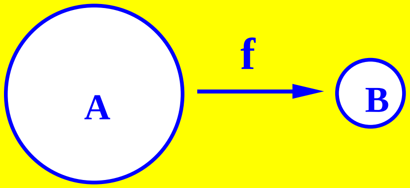
∃ x,y ε A s.t.

$$\sigma_H(x,y) > |H| \left(\frac{1}{|B|} - \frac{1}{|A|} \right)$$

proof

let a = |A| , b = |B|

By counting, we can show



$$\sigma_f(A,A) \ge b(\tfrac{a}{b} - 1)^2 \ge \tfrac{a^2}{b} - a$$

Thus

$$\sigma_H\,(A,A) \geq a^2\ |H| \left(\frac{1}{b} - \frac{1}{a}\right)$$

By the pidgeon hole principle

∃ x,y ε A s.t

$$\sigma_H\,(x,y) \geq |H| \left(\frac{1}{b} - \frac{1}{a}\right)$$

note

in most applications, *|A| >> |B|*

and then any *universal* $_2$ *class has* asymptotically a *minimum number of conflicts*

Proposition 2: Let x ε A, S ⊆ A

For f chosen randomly from a universal $_2$ class H of hash functions, the expected number of colisions is

$$\sigma_f\,(x,S) \leq \frac{|S|}{|B|}$$

proof

$$E(\sigma_f\,(x,S)) = \frac{1}{|H|} \sum_{f\,\varepsilon\,H} \sigma_f\,(x,S)$$

$$= \frac{1}{|H|} \sum_{y\,\varepsilon\,S} \sigma_H\,(x,y) \quad \text{by definition}$$

$$\leq \frac{1}{|H|} \sum_{y\,\varepsilon\,S} \frac{|H|}{|B|} \quad \text{by definition of universal}_2$$

$$= \frac{|S|}{|B|}$$

application

associative memory storage of |S|
keys onto |B| linked lists.

Given key x ε A, store x in list f(x)

Proposition 2 implies each list has expected

$$\mathit{length} \le \frac{|S|}{|B|} = 0(1) \text{ if } |B| \ge |S|$$

Gives 0(1) time for STORE, RETRIEVE,
and DELETE operations

Proposition 3

Let R be a sequence of requests with k insertion operations into an associative memory.

If f is chosen at random from set of universal$_2$ class H, the expected

total cost of all k searches is

$$\le |R| \left(1 + \frac{k}{|B|}\right).$$

proof

There are |R| total search ops,
and each takes by Proposition 2 expected

$$\text{time} \le 1 + \frac{k}{|B|}.$$

note

if |B| ≥ k, then expected total

time is O(|R|).

Bounds on distribution of $\sigma_f\ (x,S)$

Proposition 4 Let $x \in A,\ S \subset A$

Let μ = expected value of $\sigma_f\ (x,S)$

For f chosen randomly from universal $_2$ set of functions H,

$$\text{Prob}\ (\sigma_f\ (x,S) > t\cdot\mu) < \frac{1}{t}$$

proof

immediate from Markov bound

improved bounds on probability:

$$\text{prob} \le \frac{11}{t^4} \quad \text{for universal hash fns. } H_2,\ H_3$$

(using 2nd and 4th moments of prob. distribution.)

H = universal$_2$ set of hash functions.

E_1= Expected cost of *random* set of k requests using a *worst case* function f in H *(random input)*

E_2= Expected cost of *worst case* set of k requests using a *random* function f in H *(randomized algorithm)*

Prop 5 $E_1 \geq (1-\varepsilon) E_2$ where $\varepsilon = \frac{|B|}{|A|}$

proof

Let a = |A|, b = |B|.

Prop 2 implies $E_2 \leq 1 + \frac{|S|}{b}$

Suppose S is chosen randomly. for x, y ε S,

$$E(\sigma_f(x,y)) = \frac{1}{a^2}\sigma_f(A,A)$$

$$\geq \frac{1}{a^2}\left| a^2\left(\frac{1}{b}-\frac{1}{a}\right)\right| \text{by Prop 1}$$

$$\geq \left(\frac{1}{b}-\frac{1}{a}\right)$$

So $E_1 \geq 1 + E(\sigma_f(x,S))$

$$\geq 1 + |S|\left(\frac{1}{b}-\frac{1}{a}\right)$$

Example of Universal $_2$ *Class*

Set of Keys Table

Let A = {0,1, ... , a-1} Set of Keys
B = {0,1, ... , b-1} Table

Let p be a prime ≥ a

Zp = {0,1, ... , p-1} = number field mod p

define $g : Z_p \to B$ s.t.
$g(x) = x \bmod b$

define for n, m ε Z_p with $m \neq o$,
$h_{n,m}: A \to Z_p$
with $h_{n,m}(x) = (mx+n) \bmod p$

define $f_{n,m}: A \to B$ s.t. $f_{n,m}(x) = g(h_{m,n}(x))$

$$H_1 = \{f_{m,n} \mid m,n \in Z_p,\ m \neq o\}$$

Claim: H_1 is universal $_2$

Lemma

for distinct x, y εA,

$$\sigma_{H_1}(x,y) = \sigma_g(Z_p , Z_p)$$

proof

$$\sigma_g (Z_p , Z_p) = |\{(r,s) \mid r,s \ \varepsilon\ Z_p,\ r \neq s,\ g(r) = g(s)\}|$$

Observe that the linear equations:

xm + n = r (mod p)

ym + n = s (mod p)

have *unique* solutions in Z_p

So (r,s) = $(h_{m,n}(x) , h_{m,n}(y))$ then

($f_{m,n}(x) = f_{m,n}(y)$ if and only if g(r) = g(s))

$\sigma_H(x,y)$ is the number of such pairs in

$(r,s)\ \varepsilon\ \sigma_g(Z_p , Z_p)$

Theorem

H_1 is universal $_2$

proof **Let $n_i = |\{t\ \varepsilon\ Z_p \mid g(t) = i\}|$**

By definition of g(x) = x mod b,

$$\Rightarrow \quad n_i \leq \frac{p-1}{b} + 1$$

For any given r, the number of s

where s ≠ r and g(r) = g(s) is

$$\sigma_g (r , Z_p) \leq \frac{p-1}{b}.$$

But there are p choices of r,

$$\text{so} \quad p \cdot \left(\frac{(p-1)}{b}\right) \geq \sigma_g (Z_p , Z_p)$$

$$= \sigma_{H_1}(x,y) \text{ by Lemma}$$

(Also note $\sigma_H(x,x) = 0$)

Hence $\sigma_{H_1}(x,y) \leq \frac{|H_1|}{b}$ since $|H_1| = p(p-1)$

so H_1 is universal $_2$

Universal Hash Fns on *Long* keys

Given class of hash functions H,
define hash functions $J = \{h_{f,g} \mid f,g \in H\}$

where $h_{f,g}(x_1, x_2) = f(x_1) \underset{\uparrow}{\oplus} g(x_2)$

exclusive or

Theorem Suppose $B = \{0,1,\dots,b=1\}$ where b is a power of 2. Suppose this class of fns $A \longrightarrow B$

$$\exists \text{ real } r \forall i \varepsilon B \forall x_1, y_1 \varepsilon A,\ x_1 \neq y_1$$
$$\Rightarrow \left|\left\{ f \varepsilon H \mid f(x_1) \oplus f(y_1) = i \right\}\right| \leq r|H|$$

Then $\forall x, y \in (A \times A)$, $x \neq y$
$$\left|\left\{ h \varepsilon J \mid h(x) \oplus h(y) = i \right\}\right| \leq r|H|$$

Proof for $x = (x_1, x_2)$, $y = (y_1, y_2)$ in $A \times A$

$i \varepsilon B$ then $\left|\left\{ h \varepsilon J \mid h(x) \oplus h(y) = i \right\}\right|$

$$= \left|\left\{ f, g \varepsilon H \mid f(x_1) \oplus g(x_2) \oplus f(y_1) \oplus g(y_2) = i \right\}\right|$$

$$= \sum_{y \varepsilon H} \left|\left\{ f \varepsilon H \mid f(x_1) \oplus f(y_1) = i \oplus g(x_2) \oplus g(y_2) \right\}\right|$$

$$\leq \left|\left\{ f \in H \mid f(x_1) \oplus f(y_1) = i \right\}\right| \leq r|H|$$

example H_1 with $m = 0$ gives J with $r = \dfrac{1}{|B|}$ universal!

Universal$_2$ Hashing *with out Multiplication*

A = set of d digit numbers base α so, $|A| = \alpha^d$

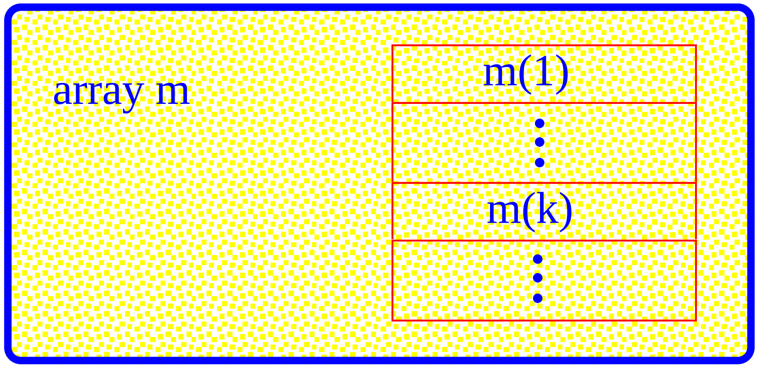
B = set of binary numbers length j

M = arrays of length $d \cdot \alpha$,
with elements in B

$\forall m \; \varepsilon \; M$ let m(k) = kth element of array m

$\forall x \; \varepsilon \; A$ let x_k = kth digit of x base α

definition $f_m(x) = m(x_1+1) \oplus m(x_1+x_2+2) \oplus \dots \oplus m\left(\sum_{k=1}^{d} x_k + k \right)$



Theorem

$H_2 = \{ f_m | m \; \varepsilon \; M \}$ is universal$_2$

proof for x, y ε A,

let $f_m(x) = r_1 \oplus r_2 \oplus \dots \oplus r_s$ rows of m

$f_m(y) = r_{s+1} \oplus \dots \oplus r_t$

Then $f_m(x) = f_m(y)$ iff $r_1 \oplus \dots \oplus r_t = \overline{o}$

But if $x \neq y \Rightarrow \exists$ k s.t. r_k in only one of $f_m(x)$, $f_m(y)$

so $\left(f_m(x) = f_m(y) \text{ iff } r_k = \bigoplus_{i \neq k} r_i \right)$

But there are only |B| possibilities for row r_k

so x,y will collide for $\frac{1}{|B|}$ of fns $f_m \; \varepsilon \; H_2$

Hence H_2 is universal$_2$

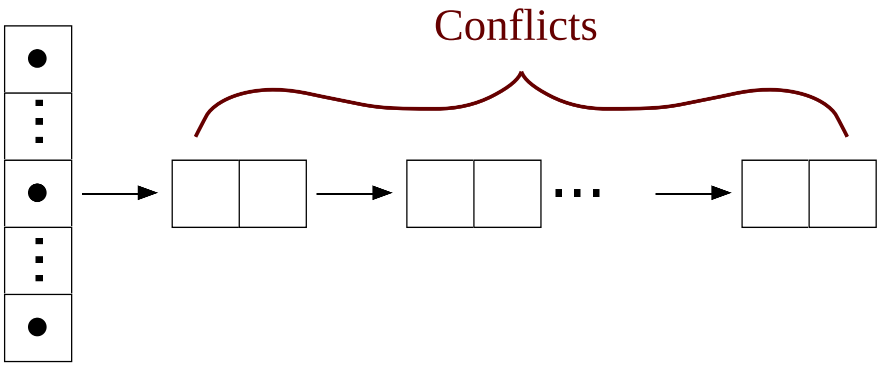
Analysis of Hashing

for Uniform Random Hash fn

load factor $\alpha = \dfrac{\text{\# of keys hashed}}{\text{\# of indicies in Hash Table}}$

Hashing with Chaining

keep list of conflicts at each index



length is ***binomial*** variable

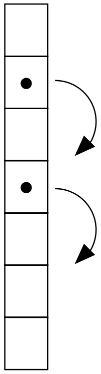
expected length = α

Expected Time Cost per hash = $O(1+\alpha)$

By Chernoff Bounds, with high likelyhood
time cost per hash $\leq O(\alpha \log(\#\ \text{keys}))$

Open Address Hashing
(With Uniform Random Hash fn)

Resolve conflicts by applying another hash function



α = load factor = prob. of occupied hash address

rehashes as geometric variable

expected hash time = $\frac{1}{1-\alpha}$ = $1+\alpha+\alpha^2+\ldots$