Universal Hashing

Luke Zhou

May 28, 2018

Hashing

Hashing is efficient.

- $\Theta(n)$ storage
- $O(1)$ time cost for all dictionary operations in average (if $n = O(m)$)

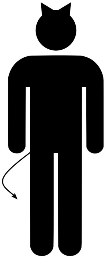
Hashing

Hashing is efficient.

- $\Theta(n)$ storage
- $O(1)$ time cost for all dictionary operations in average (if $n = O(m)$)

Hashing is not safe.

- $\Theta(n)$ time cost for search in worst case



Malicious adversary

- How can you defeat your adversary?

How can you defeat your adversary?
The answer is:

- **Randomness**

i.e. choose the hash function *randomly* in a way that is *independent* of the keys that are actually going to be stored.

- **Universal Hashing**

Analysis (informal)

Informal:

- Like randomized algorithms in Chapter 5
- Your adversary can no longer make a difference

Analysis (informal)

Informal:

- Like randomized algorithms in Chapter 5
- Your adversary can no longer make a difference
- But your luck will

Analysis (formal)

What property should the collection of hash functions have (to be useful)?

Definition

Let $\mathscr{H}$ be a finite collection of hash functions that map a given universe U of keys into the range $\{0, 1, ..., m-1\}$. Such a collection is said to be **universal** if: for each pair of distinct keys $k, l \in U$, the number of hash functions $h \in \mathscr{H}$ for which $h(k) = h(l)$ is at most $|\mathscr{H}|/m$.

Analysis (formal)

Definition

Let $\mathscr{H}$ be a finite collection of hash functions that map a given universe U of keys into the range $\{0, 1, ..., m-1\}$. Such a collection is said to be **universal** if: for each pair of distinct keys $k, l \in U$, the number of hash functions $h \in \mathscr{H}$ for which $h(k) = h(l)$ is at most $|\mathscr{H}|/m$.

- Why defined in this way?

Analysis (formal)

Definition

Let $\mathscr{H}$ be a finite collection of hash functions that map a given universe U of keys into the range $\{0, 1, ..., m-1\}$. Such a collection is said to be **universal** if: for each pair of distinct keys $k, l \in U$, the number of hash functions $h \in \mathscr{H}$ for which $h(k) = h(l)$ is at most $|\mathscr{H}|/m$.

Why defined in this way?

The ideal uniform random hashing:

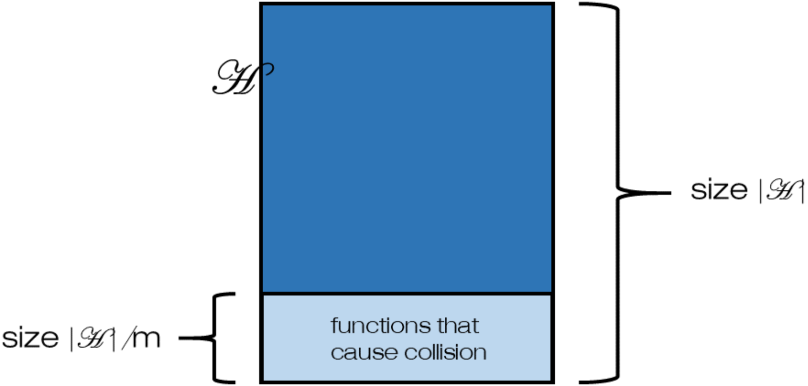
- a collision of two keys has probability $1/m$

Now we have an approximate (weaker) random hashing:

- a collision of two keys has probability $\frac{|\mathscr{H}|/m}{|\mathscr{H}|} = 1/m$

(ϵ-almost universality$<$uniform difference property$<$pairwise independence/strong universality)

Analysis (formal)



Analysis (formal)

Theorem 11.3

$h \in \mathscr{H}$ chosen randomly, hashing n keys $\to T$ (chaining), then the expected length of the list that the key k hashes has bounds:

$$E[n_{h(k)}] \le \begin{cases} \alpha & \text{key } k \text{ is not in the table,} \\ 1 + \alpha & \text{key } k \text{ is in the table.} \end{cases}$$

Proof Page 1

For each pair k and l of distinct keys, define the indicator random variable $X_{kl} = I\{h(k) = h(l)\}$.
By definition, $Pr\{h(k) = h(l)\} \leq 1/m$. Thus, $E[X_{kl}] \leq 1/m$.

Analysis (formal)

Proof Page 1

For each pair k and l of distinct keys, define the indicator random variable $X_{kl} = I\{h(k) = h(l)\}$.
By definition, $Pr\{h(k) = h(l)\} \leq 1/m$. Thus, $E[X_{kl}] \leq 1/m$.
Define Y_k, the number of keys other than k that hash to the same slot as k, then

$$
\begin{aligned}
E[Y_k] &= E[\sum_{l \in T, l \neq k} X_{kl}] \\
&= \sum_{l \in T, l \neq k} E[X_{kl}] \\
&\leq \sum_{l \in T, l \neq k} \frac{1}{m}
\end{aligned}
$$

Proof Page 2

$$E[Y_k] \leq \sum_{l \in T, l \neq k} \frac{1}{m}$$
$$= \begin{cases} n \cdot \frac{1}{m} = \alpha & \text{key } k \text{ is not in the table,} \\ (n-1) \cdot \frac{1}{m} < \alpha & \text{key } k \text{ is in the table.} \end{cases}$$

Analysis (formal)

Proof Page 2

$$E[Y_k] \leq \sum_{l \in T, l \neq k} \frac{1}{m}$$

$$= \begin{cases} n \cdot \frac{1}{m} = \alpha & \text{key } k \text{ is not in the table,} \\ (n-1) \cdot \frac{1}{m} < \alpha & \text{key } k \text{ is in the table.} \end{cases}$$

Thus, we can conclude that

$$E[n_{h(k)}] = \begin{cases} E[Y_k] \leq \alpha & \text{key } k \text{ is not in the table,} \\ 1 + E[Y_k] \leq 1 + \alpha & \text{key } k \text{ is in the table.} \end{cases}$$

$\square$

Analysis (formal)

Theorem 11.3

$h \in \mathscr{H}$ chosen randomly, hashing n keys $\to T$ (chaining), then the expected length of the list that the key k hashes has bounds:

$$E[n_{h(k)}] \leq \begin{cases} \alpha & \text{key } k \text{ is not in the table,} \\ 1+\alpha & \text{key } k \text{ is in the table.} \end{cases}$$

This guarantees the (average) performance of the hashing.

Performance (Corollary 11.4)

Using universal hashing and collision resolution by chaining in an initially empty table with m slots, further assuming that $n = O(m)$, then we only need

$$O(1) \text{ time cost}$$

for all dictionary operations in average.

- Then, how to construct one?

A "particularly elegant" construction:

Construction

Let m be prime. Decompose key k into $r+1$ digits.
$k =< k_0, k_1, ..., k_r >$ where $0 \leq k_i \leq m-1$. (base m)
Pick $a =< a_0, a_1, ..., a_r >$ where $0 \leq a_i \leq m-1$.
Define

$$h_a(k) = \left(\sum_{i=0}^{r} a_i k_i \right) \bmod m$$

(dot product + modulo m)
Then here

$$|\mathscr{H}| = m^{r+1}$$

Construction

Theorem

The class of hash functions $\mathscr{H}$ is universal.

Proof Page 1

Pick two distinct keys arbitrarily:

$$x =< x_0, x_1, ..., x_r >$$

$$y =< y_0, y_1, ..., y_r >$$

They differ in at least one digit, without loss of generality, position zero.

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

Construction

Proof Page 2

$$
\begin{aligned}
& h_a(x) = h_b(y) \\
\Leftrightarrow & \sum_{i=0}^{r} a_i x_i \equiv \sum_{i=0}^{r} a_i y_i \pmod{m} \\
\Leftrightarrow & \sum_{i=0}^{r} a_i (x_i - y_i) \equiv 0 \pmod{m} \\
\Leftrightarrow & a_0(x_0 - y_0) + \sum_{i=1}^{r} a_i (x_i - y_i) \equiv 0 \pmod{m} \\
\Leftrightarrow & a_0(x_0 - y_0) \equiv -\sum_{i=1}^{r} a_i (x_i - y_i) \pmod{m}
\end{aligned}
$$

Construction

Lemma (number theory fact)

Let m be prime.
For any $z \in \mathbb{Z}_m$ (integers mod m) such that $z \not\equiv 0$,
there $\exists$ unique $z^{-1} \in \mathbb{Z}_m$ such that $z \cdot z^{-1} \equiv 1 \pmod{m}$.

Construction

Lemma (number theory fact)

Let m be prime.
For any $z \in \mathbb{Z}_m$ (integers mod m) such that $z \not\equiv 0$,
there $\exists$ unique $z^{-1} \in \mathbb{Z}_m$ such that $z \cdot z^{-1} \equiv 1 \pmod{m}$.

e.g.

Ex. $m = 7$:

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

Proof Page 3

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

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

Proof Page 3

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

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

That means, for any choice of $a_1, a_2, ..., a_r$,
exactly 1 choice of the m choices for a_0 causes $h_a(x) = h_a(y)$,
and $h_a(x) \neq h_a(y)$ for other $m-1$ choices for a_0.

Proof Page 4

Thus, the number of $h_a \in \mathscr{H}$ such that $h_a(x) = h_a(y)$ is

$$\underbrace{m \cdot m \cdot \ldots \cdot m}_{\text{there are } r \text{ factors, for } a_1 \text{ to } a_r} \cdot \underbrace{1}_{\text{for } a_0}$$

$= m^r = \frac{|\mathscr{H}|}{m}$. □

Thanks

References:

- MIT OpenCourseWare 6.046J
- Universal_hashing of Wikipedia