

Hash Tables: Hash Functions

Michael Levin

Higher School of Economics

Data Structures

<http://bit.ly/algospecialization>

Outline

- 1 Chain Length for Universal Family
- 2 Universal Family for Integers

Math Used

- Probabilities
- Expectation and linearity

Reminder: Universal Family

Definition

Let U be the **universe** — the set of all possible keys. A set of hash functions $\mathcal{H} : U \rightarrow \{0, 1, \dots, m - 1\}$ with **cardinality** m is called a **universal family** if for any two keys $x, y \in U, x \neq y$ the probability

$$\Pr[h(x) = h(y)] \leq \frac{1}{m}$$

Reminder: Meaning of Probability

The probability

$$Pr[h(x) = h(y)]$$

is taken over the random choice of a hash function h from the set \mathcal{H} .

Reminder: Reformulation

Equivalent definition: for any two keys $x, y \in U, x \neq y$ at most $\frac{1}{m}$ of all hash functions $h \in \mathcal{H}$ produce a collision $h(x) = h(y)$.

Reminder: Load Factor

Definition

Let T be a hash table of size m which stores n keys. $\alpha = \frac{n}{m}$ is called the **load factor** of this hash table.

Linearity of Expectation

Lemma

For any finite list of random variables

X_1, X_2, \dots, X_k and $Y = X_1 + X_2 + \dots + X_k$,
 $E(Y) = E(X_1) + E(X_2) + \dots + E(X_k)$.

Theorem

Suppose h is selected at random from a universal family \mathcal{H} and is used to hash n keys into hash table T of size m giving load factor α . Then for any key k the expected length $E[n_{h(k)}]$ of the chain in T to which k is hashed is at most $1 + \alpha$.

Proof

- Fix key k

Proof

- Fix key k
- For any other key l , define random variable

$$X_{kl} = \begin{cases} 1, & \text{if } h(k) = h(l) \\ 0, & \text{otherwise} \end{cases}$$

Proof

- Fix key k
- For any other key l , define random variable

$$X_{kl} = \begin{cases} 1, & \text{if } h(k) = h(l) \\ 0, & \text{otherwise} \end{cases}$$

- $E(X_{kl}) = 0 + 1 \times \Pr[h(k) = h(l)] \leq \frac{1}{m}$

Proof

■ Number of collisions $Y_k = \sum_{l \neq k, l \in T} X_{kl}$

Proof

- Number of collisions $Y_k = \sum_{l \neq k, l \in T} X_{kl}$
- Chain length $n_{h(k)} = 1 + Y_k$

Proof

- Number of collisions $Y_k = \sum_{l \neq k, l \in T} X_{kl}$
- Chain length $n_{h(k)} = 1 + Y_k$
- $E(Y_k) = \sum_{l \neq k, l \in T} E(X_{kl})$

Proof

- Number of collisions $Y_k = \sum_{l \neq k, l \in T} X_{kl}$
- Chain length $n_{h(k)} = 1 + Y_k$
- $E(Y_k) = \sum_{l \neq k, l \in T} E(X_{kl}) \leq \sum_{l \neq k, l \in T} \frac{1}{m} \leq$
 $\leq \frac{n}{m} = \alpha$

Proof

- Number of collisions $Y_k = \sum_{l \neq k, l \in T} X_{kl}$
- Chain length $n_{h(k)} = 1 + Y_k$
- $E(Y_k) = \sum_{l \neq k, l \in T} E(X_{kl}) \leq \sum_{l \neq k, l \in T} \frac{1}{m} \leq$
 $\leq \frac{n}{m} = \alpha$
- $E(n_{h(k)}) = 1 + E(Y_k) \leq 1 + \alpha$



Corollary

Corollary

Using universal hashing and chaining scheme in a hash table of size m , it takes expected time $\Theta(n)$ to perform n operations of insertion, deletion, and search if there are $O(m)$ insertion operations. Thus, operations with the hash table run in amortized $O(1)$ time on average.

Corollary Proof

Proof

- $O(m)$ insertions, so
 $n = O(m), \alpha = O(1)$

Corollary Proof

Proof

- $O(m)$ insertions, so
 $n = O(m), \alpha = O(1)$
- $1 + \alpha = O(1)$

Corollary Proof

Proof

- $O(m)$ insertions, so
 $n = O(m), \alpha = O(1)$
- $1 + \alpha = O(1)$
- Expected running time of each operation is $O(1)$

Corollary Proof

Proof

- $O(m)$ insertions, so
 $n = O(m), \alpha = O(1)$
- $1 + \alpha = O(1)$
- Expected running time of each operation is $O(1)$
- Expected running time of n operations is $\Theta(n)$

Conclusion

- Proven upper bound $1 + \alpha$ on the expected chain length
- Proven $O(1)$ amortized expected running time for operations with a hash table using universal family and chaining

Outline

- ① Chain Length for Universal Family
- ② Universal Family for Integers

Math Used

- Properties of prime numbers
- Properties of modulo arithmetics
- One-to-one correspondence
- Upper integral part $\lceil a \rceil$ properties
- Probabilities

Theorem

Set of functions

$$\mathcal{H}_p = \{h_p^{a,b}(x) = ((ax + b) \bmod p) \bmod m\}$$

with parameters

$$a, b : 1 \leq a \leq p - 1, 0 \leq b \leq p - 1$$

and prime p is a universal family for

$$U = \{0, 1, \dots, p - 1\}.$$

Lemma

For a fixed hash function $h = h_p^{a,b}$ from \mathcal{H}_p and keys $x, y \in U, x \neq y$ the values

$$r = (ax + b) \bmod p$$

and

$$s = (ay + b) \bmod p$$

are different.

Proof by Contradiction

Proof

$$r = s \Rightarrow (ax + b) \equiv (ay + b) \pmod{p} \Rightarrow$$

Proof by Contradiction

Proof

$$r = s \Rightarrow (ax + b) \equiv (ay + b) \pmod{p} \Rightarrow$$

$$a(x - y) \equiv 0 \pmod{p} \Rightarrow p \text{ divides } a(x - y)$$

Proof by Contradiction

Proof

$$r = s \Rightarrow (ax + b) \equiv (ay + b) \pmod{p} \Rightarrow$$

$$a(x - y) \equiv 0 \pmod{p} \Rightarrow p \text{ divides } a(x - y)$$

$$1 \leq a \leq p - 1 \Rightarrow p \text{ divides } (x - y)$$

Proof by Contradiction

Proof

$$r = s \Rightarrow (ax + b) \equiv (ay + b) \pmod{p} \Rightarrow$$

$$a(x - y) \equiv 0 \pmod{p} \Rightarrow p \text{ divides } a(x - y)$$

$$1 \leq a \leq p - 1 \Rightarrow p \text{ divides } (x - y)$$

$$0 \leq x, y < p, p \text{ divides } (x - y) \Rightarrow x = y \quad \square$$

Corollary

There are no collisions for

$$h(x) = (ax + b) \bmod p,$$

before taking the value mod m .

Lemma

For fixed keys $x \neq y$, there is one-to-one correspondence between pairs

$(a, b), 1 \leq a \leq p-1, 0 \leq b \leq p-1$ and
 $(r, s), 0 \leq r, s \leq p-1, r \neq s$

Proof

Different (a, b) generate different (r, s) :

Proof

Different (a, b) generate different (r, s) :

$$a = ((r - s)((x - y)^{-1} \bmod p) \bmod p,$$

$$b = (r - ax) \bmod p$$

Proof

Different (a, b) generate different (r, s) :

$$a = ((r - s)((x - y)^{-1} \bmod p) \bmod p,$$

$$b = (r - ax) \bmod p$$

$$r = r', s = s' \Rightarrow a = a', b = b'$$

Proof

- Different (a, b) generate different (r, s)
- $p(p - 1)$ total pairs (a, b)
- $p(p - 1)$ total pairs (r, s)
- Thus one-to-one correspondence □

Corollary

If x and y , $x \neq y$ are some keys, any $h \in \mathcal{H}_p$ is chosen at random with equal probability $\frac{1}{p(p-1)}$, then each pair of values

$$(r, s) = ((ax + b) \bmod p, (ay + b) \bmod p)$$

happen with equal probability $\frac{1}{p(p-1)}$.

Proof

- There is one-to-one correspondence between (a, b) and (r, s)

Proof

- There is one-to-one correspondence between (a, b) and (r, s)
- Probability of any pair (a, b) is $\frac{1}{p(p-1)}$

Proof

- There is one-to-one correspondence between (a, b) and (r, s)
- Probability of any pair (a, b) is $\frac{1}{p(p-1)}$
- So probability of any (r, s) is $\frac{1}{p(p-1)}$ □

Proof of the Theorem

Proof

$$Pr[h(x) = h(y)] = Pr[r \bmod m = s \bmod m]$$

Proof of the Theorem

Proof

$$\Pr[h(x) = h(y)] = \Pr[r \bmod m = s \bmod m]$$

Each pair (r, s) has probability $\frac{1}{p(p-1)}$

Proof of the Theorem

Proof

For each $r \in [0, p - 1]$, there are at most $\lceil \frac{p}{m} \rceil - 1$ such s that $s \neq r$ and $r \bmod m = s \bmod m$:

Proof of the Theorem

Proof

For each $r \in [0, p-1]$, there are at most $\lceil \frac{p}{m} \rceil - 1$ such s that $s \neq r$ and $r \bmod m = s \bmod m$:

$$0, 1, \dots, \check{m}, \dots, 2\check{m}, \dots, (\lceil \frac{p}{m} \rceil - 1)m, \dots$$

Proof of the Theorem

Proof

Proof of the Theorem

Proof

$$\begin{aligned} Pr[r \bmod m = s \bmod m] &\leq \sum_{r=0}^{p-1} \frac{\lceil \frac{p}{m} \rceil - 1}{p(p-1)} = \\ \frac{\lceil \frac{p}{m} \rceil - 1}{(p-1)} &\leq \frac{\frac{p+m-1}{m} - 1}{(p-1)} = \frac{p-1}{m(p-1)} = \frac{1}{m} \end{aligned}$$

Proof of the Theorem

Proof

$$Pr[r \bmod m = s \bmod m] \leq \sum_{r=0}^{p-1} \frac{\lceil \frac{p}{m} \rceil - 1}{p(p-1)} =$$

$$\frac{\lceil \frac{p}{m} \rceil - 1}{(p-1)} \leq \frac{\frac{p+m-1}{m} - 1}{(p-1)} = \frac{p-1}{m(p-1)} = \frac{1}{m}$$

$$Pr[h(x) = h(y)] \leq \frac{1}{m}$$



Conclusion

- Proven universal family for integers
- Proven $1 + \alpha$ bound for expected chain length
- Proven $O(1)$ amortized expected running time of hash table operations