CS6033 Lecture 3
Slides/Notes

Review of Math Background in Probability; Hash Tables (Notes, Ch 11 (skip Secs. 11.3.4, 11.3.5 and 11.5))

By Prof. Yi-Jen Chiang
CSE Dept., Tandon School of Engineering
New York University

1

Abstract Data Type (ADT): Dictionary D

Support 3 operations:

1. Insert (X, D) (X: item with a key)
2. Search (K, D) (K: key)
3. Delete (K, D): Delete the item with key K from D.

Hash Table: Randomized Algorithms (Using probabilistic analysis)

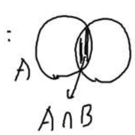
Review of Math Background in Probability

1. Boole's Inequality (Union Bound).

① Let A, B be 2 events.

$Pr[A \cup B] \le Pr[A] + Pr[B]$.

Pf:



$Pr[A \cup B] = Pr[A] + Pr[B] - Pr[A \cap B] \le Pr[A] + Pr[B]$.

② Let $A_1, A_2 .. A_n$ be n events.

$Pr[A_1 \cup \cdots \cup A_n] \le \sum_{i=1}^{n} Pr[A_i]$

2

2. Linearity of Expectation

Def: Let X be a random variable. Expected value of X, $E[X]$, is defined as

$$E[X] = \sum_{x} x \cdot \Pr[X=x]$$

Intuition: weighted sum of value where the weights are probabilities

e.g. Exam score: 95, 92, 80, 75, 71

Average: $(95+92+80+75+71)/5$

$= \frac{1}{5} \cdot 95 + \frac{1}{5} \cdot 92 + \cdots$

e.g. $\underbrace{95, 95}_{\Pr = \frac{2}{5}}$, $\underbrace{80}_{\frac{1}{5}}$, $\underbrace{75, 75}_{\frac{2}{5}}$.

Average: $95 \cdot \frac{2}{5} + 80 \cdot \frac{1}{5} + 75 \cdot \frac{2}{5}$.

3

Linearity of Expectation

① Let X, Y be 2 random variables NOT necessarily independent (i.e. they can be dependent.)

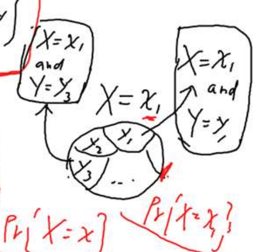
$$E[X+Y] = E[X] + E[Y].$$

Pf: $E[X+Y] = \sum_{x,y} \underline{(x+y)} \cdot \Pr[X=x \text{ and } Y=y]$.

$$= \underbrace{\boxed{\sum_{x,y} x \cdot \Pr[X=x \text{ and } Y=y]}}_{A} + \underbrace{\boxed{\sum_{x,y} y \cdot \Pr[X=x \text{ and } Y=y]}}_{B}$$

B // $E[Y]$ by the same process.

A = $\sum_{x} \sum_{y} x \cdot \Pr[X=x \text{ and } Y=y] = \sum_{x} x \left(\underbrace{\sum_{y} \Pr[X=x \text{ and } Y=y]}_{= \Pr[X=x]} \right)$



$= \sum_{x} x \cdot \Pr[X=x]$

$= E[X]$.

$A = E[X]$, $B = E[Y]$.

$E[X+Y] = A + B = E[X] + E[Y]$ ※

4

② Let $X_1, X_2 \cdots X_n$ be n random variables. (they can be dependent)

$$E[X_1 + X_2 + \cdots + X_n] = E[X_1] + E[X_2] + \cdots + E[X_n].$$

Pf: $E[X_1 + X_2 + \cdots + X_n] = E[\underbrace{(X_1 + \cdots + X_{n-1})} + \underbrace{X_n}] = \underbrace{E[X_1 + \cdots + X_{n-1}]} + E[X_n]$.

$= E[X_1 + \cdots + X_{n-2}] + E[X_{n-1}] + E[X_n] = \cdots = E[X_1] + E[X_2] + \cdots + E[X_n]$.

3. Bernoulli Trial: Flip a coin (with $Pr[\text{Head}] = p$ and $Pr[\text{Tail}] = q = 1-p$) many times, each time is indept. of the others.

① Expected # of flips to get the first head $= \frac{1}{p}$. $\left(\text{eg. } p = \frac{1}{2} \Rightarrow \frac{1}{p} = 2\right)$.

5

Pf: M1: Let X be random variable for the # flips to get the first head.

$$E[X] = \sum_{i=1}^{\infty} i \cdot Pr[X = i]$$

$\left(Pr\{X = i\} = Pr\{\text{first } (i-1) \text{ flips are tails \& the last flip is head}\} = q^{i-1} p. \right)$

$$= \sum_{i=1}^{\infty} i \cdot (q^{i-1} \cdot p) = S$$

$S = 1 \cdot p + 2 \cdot qp + 3 \cdot q^2 p + 4 q^3 p + \cdots$

$-)\ qS = \quad 1 \cdot qp + 2 \cdot q^2 p + 3 q^3 p + \cdots$

$(1-q) S = 1 \cdot p + 1 \cdot qp + 1 \cdot q^2 p + 1 \cdot q^3 p + \cdots$

$= p(1 + q + q^2 + q^3 + \cdots) = p \cdot \frac{1}{1-q} = p \cdot \frac{1}{p} = 1$

$\boxed{(1-q) S = 1}$

$\Rightarrow S = \frac{1}{1-q} = \frac{1}{p}$.

$S = E[X] = \frac{1}{p}$. #

M2: Let E be the desired expected # of flips.

* MA: $E = 1 + 0 \cdot p + E \cdot q \leftrightarrow$ *(first flip is Head: $Pr = p$. 0 additional flips; Tail: $Pr = q$. E)

$\boxed{(1-q) E = 1} \Rightarrow E = \frac{1}{1-q} = \frac{1}{p}$ #

6

*MB : $E = p \cdot (1+0) + q(1+E)$ *

\# total flips is $\begin{cases} (1+0) & \text{if } \underbrace{\text{first flip is Head}}_{P_r = p} \\ (1+E) & \text{if } \underbrace{\text{first flip is Tail}}_{P_r = q} \end{cases}$

$E = p + q + qE$

$= 1 + qE \Rightarrow E = \frac{1}{1-q} = \frac{1}{p}$ ※ $(p+q=1)$

$$A = 1 + q + q^2 + \cdots + -$$
$$-) \; qA = \quad q + q^2 + \cdots -$$
$$(1-q)A = 1 \Rightarrow A = \frac{1}{1-q}.$$

② Expected # of flips to get the K-th head: X

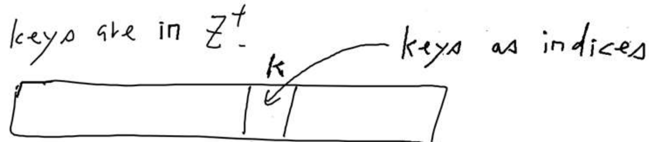
1st Head	2nd Head	--	k-th Head.
OOO~●	O---●	---	OOO--●
T T T H	T-- H		TT~ H
X_1	X_2		X_K.

Def : X_i : # flips to get the i-th Head after getting the $(i-1)$st Head.

$X = X_1 + X_2 + \cdots + X_K$ $\quad E[X] = E[X_1] + E[X_2] + \cdots + E[X_K] = \frac{1}{p} \cdot K = \frac{K}{p}$ ※

7

Hash Table $\quad$ Array: keys are in Z^+.



Keys are from U, where U is a large set of possible key values.

n items. If we use an array: $\boxed{\text{O(1) worst-case time}}$ → for insert, delete, search Good

But we need memory space $|U| >> n$. Bad!

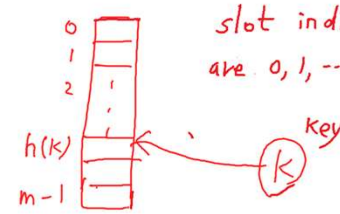
Hash Table: Use memory space $m = O(n)$ and retain $O(1)$ time for insert, delete, search.

→ Typically expected time [worst-case time can be $\Theta(n)$]

[Look at details for each specific approach.]

8

Main Idea: Use a hash table of m slots



slot indices are $0, 1, \cdots, (m-1)$.

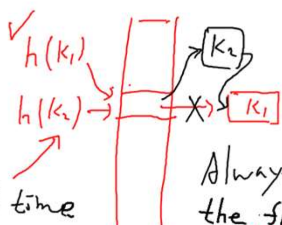
Use a hash function $h : \mathcal{U} \to \{0, 1, 2, \cdots, (m-1)\}$

For each key $k \in \mathcal{U}$. $h(k) \in \{0, 1, \cdots, (m-1)\}$

Issue: Different keys may be mapped/hashed to the same slot.

Called Collision

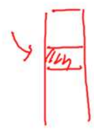
2 ways to handle collisions: M1: chaining



Insert: $O(1)$ time worst-case

Always add to the front of the list/chain

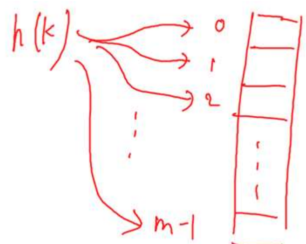
M2: Open Addressing (next class)



9

Ideal Situation/ Strong Assumption: Independent uniform hashing (Simple uniform hashing in 3rd Ed.)

Assume: For each key k. $h(k)$ goes to one of the m slots with equal prob $\frac{1}{m}$ i.e. $\Pr[h(k) = i] = \frac{1}{m}$, $\forall k \in \mathcal{U}$ and $i \in \{0, 1, \cdots, (m-1)\}$.



[Meaning: the hash function $h()$ is perfect]

(1) Expected search time for an unsuccessful search

(2) Expected search time for a successful search

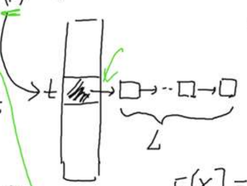
10

Under independent uniform hashing assumption.

Thm A: Expected time for an unsuccessful search is $\Theta(\alpha+1)$.

Pf: Let k be the key of the unsuccessful search.
(k is NOT any of the n items in the hash table)

Suppose $h(k)=t$



We need to spend at least $\Theta(1)$ time
$\Rightarrow \Theta(\alpha+1)$ #

Let X be a random var. for the search time.

$X = X_1 + X_2 + \cdots + X_n$

where X_i is a random var: $X_i = \begin{cases} 1 & \text{if } h(i)=h(k) \\ 0 & \text{if } h(i)\neq h(k) \end{cases}$

$\boxed{\Pr\{X_i=1\} = \frac{1}{m}.}$

$$E[X] = E[X_1]+E[X_2]+\cdots+E[X_n] = \left(1\cdot\frac{1}{m}+0\cdot\left(1-\frac{1}{m}\right)\right)\cdot n = \frac{n}{m} = \alpha$$

Recall:
- n: # items
- m: # slots in Hash Table.

Def: Load factor $\alpha = \frac{n}{m}$.

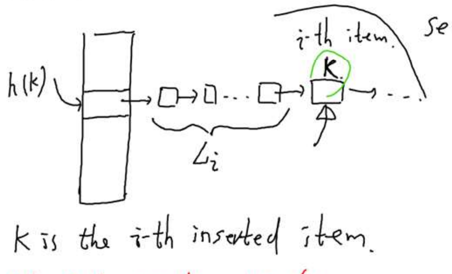
Assume: α is some constant.

11

Thm B: The expected time for a successful search is $\Theta(\alpha+1)$.

Pf: The key k is one of the n items inserted before. (searched key) and each of these n items has an equal prob of $\frac{1}{n}$ to be the searched key k. (We will range i from $1, 2, \cdots, n$ with equal prob $\frac{1}{n}$)

Let k be the i-th inserted item.



search time: $|L_i|+1$

$|L_i|$ = # items inserted after the i-th item and are hashed into $h(k)$.

$= X_{i+1} + X_{i+2} + \cdots + X_n$

X_j is random var for the j-th inserted item ℓ

$X_j = \begin{cases} 1 & \text{if } h(\ell)=h(k) \\ 0 & \text{else.} \end{cases}$ $\boxed{\Pr\{X_j=1\}=\frac{1}{m}}$ $(\ell\neq k)$

k is the i-th inserted item.

Let X be random var for search time. $X = |L_i|+1$

$$E[X] = \frac{1}{n}\sum_{i=1}^{n} E[|L_i|+1] = \frac{1}{n}\sum_{i=1}^{n}\left(1+E[X_{i+1}]+E[X_{i+2}]\cdots+E[X_n]\right) = \frac{1}{n}\sum_{i=1}^{n}\left(1+\frac{1}{m}(n-i)\right)$$

12

$$E[X] = \frac{1}{n}\sum_{i=1}^{n}\left(1+\frac{n-i}{m}\right) = \frac{1}{n}\left[n+\sum_{i=1}^{n}\frac{n-i}{m}\right]$$

$$= 1 + \frac{1}{n}\sum_{i=1}^{n}\frac{1}{m}(n-i) = 1+\frac{1}{mn}\sum_{i=1}^{n}(n-i) = 1+\frac{1}{mn}\left(0+1+2+\cdots+(n-1)\right)$$

$$= 1+\frac{1}{m\cancel{n}}\cdot\frac{(n-1)\cdot\cancel{n}}{2} = 1+\frac{1}{2}\frac{n}{m}-\frac{1}{2m} = \Theta\left(1+\frac{1}{2}\alpha\right) = \Theta(1+\alpha) \quad \#$$

13

Hash Functions

* Static Hashing:

1. Division Method : $h(k) = k \bmod m$. Typicall we use a prime number for m.

2. Multiplication Method : Let A be a real #. $A \in (0,1)$ $h(k) = \lfloor \underbrace{(A \cdot k \bmod 1)}_{A \cdot k \text{, take the decimal part}} \cdot m \rfloor$

2'. Multiply-shift Method : (special case of multiplication method. See textbook)

* Random Hashing: We have a family $\mathcal{H}$ of hash functions.
For each execution (a sequence of insert, search, delete ops), we randomly select one hash function from $\mathcal{H}$ to use.

* Universal Hashing (most important one. See next) | Others: see textbook.

14

★ Universal Hashing: Use a family $\mathcal{H}$ of hash functions.

For each execution, randomly choose one hash function $h()$ from $\mathcal{H}$ (with equal prob. $\frac{1}{|\mathcal{H}|}$) to use.
(size $|\mathcal{H}|$)

Property of univ. hashing: For any pair of keys $k \neq l$, there are at most t hash functions $h()$ in $\mathcal{H}$ s.t. $h(k) = h(l)$ i.e. $h()$ hashes k, l into collision.

Goal: $\boxed{\forall \text{keys } k \neq l,\ Pr\{h(k) = h(l)\} = \frac{1}{m}}$ (i.e. $Pr\{\text{hashing into collision}\} = \frac{1}{m}$)

Pr [chosen hash function is one of the t functions to hash into collision]

$= \frac{t}{|\mathcal{H}|}$ $\Rightarrow$ We want: $\frac{t}{|\mathcal{H}|} = \frac{1}{m}$ $\Rightarrow$ take $t = \frac{|\mathcal{H}|}{m}$. #

* Using "$\boxed{\forall \text{keys } k \neq l,\ Pr\{h(k) = h(l)\} = \frac{1}{m}}$" the pfs for Thm A & Thm B carry over !!
No need for the assumption of independent uniform hashing !!

15

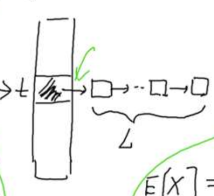
Under independent uniform hashing assumption.

Thm A: Expected time for an unsuccessful search is $\Theta(\alpha + 1)$.

Recall:
n: # items
m: # slots in Hash Table.
Def: Load factor $\alpha = \frac{n}{m}$.
Assume: α is some constant.

Pf: Let k be the key of the unsuccessful search.
(k is NOT any of the n items in the hash table)

Suppose $h(k) = t$



Let X be a random var. for the search time.

$X = X_1 + X_2 + \cdots + X_n$

where X_i is a random var: $X_i = \begin{cases} 1 & \text{if } h(i) = h(k) \\ 0 & \text{if } h(i) \neq h(k) \end{cases}$

$\boxed{Pr\{X_i = 1\} = \frac{1}{m}}$

$E[X] = E[X_1] + E[X_2] + \cdots + E[X_n] = \left(1 \cdot \frac{1}{m} + 0 \cdot \left(1 - \frac{1}{m}\right)\right) \cdot n = \frac{n}{m} = \alpha$

We need to spend at least $\Theta(1)$ time
$\Rightarrow \Theta(\alpha + 1)$ #

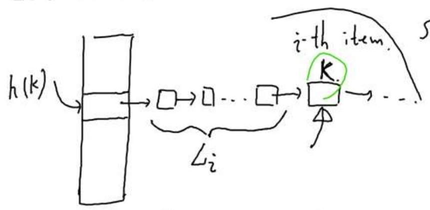
equivalent to $\boxed{\begin{array}{c}\forall \text{keys } k, l,\ k \neq l \\ Pr\{h(k) = h(l)\} = \frac{1}{m}\end{array}}$ (Property obtained from universal hashing)

16

Thm B: The expected time for a successful search is $\Theta(\alpha+1)$.

Pf: The searched key K is one of the n items inserted before. and each of these n items has an equal prob of $\frac{1}{n}$ to be the searched key k. (we will range i from $1, 2, \cdots, n$ with equal prob. $\frac{1}{n}$)

Let k be the i-th inserted item.



search time: $|L_i|+1$

$|L_i|$ = # items inserted after the i-th item and are hashed into $h(k)$.

$= X_{i+1} + X_{i+2} + \cdots + X_n$

(Property from universal hashing)

$\forall$ keys k, l, $k \neq l$ $\Pr\{h(k)=h(l)\} = \frac{1}{m}$

equiv.

X_j is random. var for the j-th inserted item l

$X_j = \begin{cases} 1 & \text{if } h(l)=h(k) \\ 0 & \text{else} \end{cases}$ $\boxed{\Pr\{X_j=1\} = \frac{1}{m}}$ $(l \neq K)$

K is the i-th inserted item.

Let X be random. var for search time. $X = |L_i|+1$

$E[X] = \frac{1}{n}\sum_{i=1}^{n} E[|L_i|+1] = \frac{1}{n}\sum_{i=1}^{n}\left(1 + E[X_{i+1}] + E[X_{i+2}] \cdots + E[X_n]\right) = \frac{1}{n}\sum_{i=1}^{n}\left(1+\frac{1}{m}(n-i)\right) = \cdots$

(The rest is the same)

17