CSI 33 LECTURE NOTES (Ojakian)

Topic 11: Hash Tables

OUTLINE
(References: 13.5)

1. Hash Tables
2. Hash Functions

---

1. The Motivation

(a) Consider typical Python Dictionary: Many keys, uncertain which will be used. How keep the space used efficient.

PROBLEM 1. *Consider trying to code a Python Dictionary from scratch. Suppose the keys are all the 5 letter words (say all CAPS). Suppose you want to represent the Dictionary as an array. Consider the following issues:*

i. How long would your array need to be to accommodate all the keys?

ii. Suppose you typically just use a small number of keys at any one time. What is wasteful in our representation?

iii. How do you get from a key to an array location quickly? (called "hash function")

2. Simple Hash Table and the Collision Issue

(a) Hash Table:

i. Set of allowed keys
ii. Array of some choosen length (can grow)
iii. Hash function: KEYS to Array Indices
iv. Mechanism for growing the array when it runs short on space
v. Mechanism for dealing with collisions?! (worry about later ...)

(b)

PROBLEM 2. *Suppose your keys are all pairs of integers* (x, y) *where* $1 \leq x, y \leq 10$.

i. How many keys are there?

ii. Suppose we use the hash function $H((x, y)) = x*y$ *mod 10. Consider a sequence of calls to the hash function, determining which cause a collision.*

iii. Make up another Hash function

(c) Making a simple Hash Table

PROBLEM 3. *Write the code for Problem 1, creating a simple Hash Table for it.*

3. Collision Issue: Chaining

PROBLEM 4. *Consider the Hash Table example in Problem 2. Choose a series of hash calls with collisions and show the resulting Hash Table, when Chaining is used to deal with collisions.*

PROBLEM 5. *What is the Theta analysis for Chaining, worst-case and "typical" case?*

PROBLEM 6. *See the program for the Chaining solution.*

4. Collision Issue: Open Addressing (like Linear Probing)

PROBLEM 7. *Consider the Hash Table example in Problem 2. Choose a series of hash calls with collisions and show the resulting Hash Table, when Open Addressing is used to deal with collisions.*

PROBLEM 8. *What is the Theta analysis for Open Addressing, worst-case and "typical" case?*

PROBLEM 9. *See the program for the Open Addressing Solution.*