HASHTABLES

David Kauchak
CS 140 – Spring 2023

1

Admin

Assignment 6 due Friday

No LCs

Mentor hours for the rest of this week:

- Wednesday, 7-9:30pm: Claire and David
- Friday, 1-3pm: Jan

2

Running time of insert and search for open addressing

Average case?

We have to make at least one probe

3

Running time of insert and search for open addressing

Average case?

What is the probability that the first probe will **not** be successful (assume uniform hashing function)?

α

4

Running time of insert and search for open addressing

Average case?

What is the probability that the first **two** probed slots will **not** be successful?

why '~'?

$\sim\alpha^2$

5

Running time of insert and search for open addressing

Average case?

What is the probability that the first **two** probed slots will **not** be successful

Technically, second probe is: $\frac{n-1}{m-1}$

$\sim\alpha^2$

6

Running time of insert and search for open addressing

Average case?

What is the probability that the first **three** probed slots will **not** be successful?

$\sim\alpha^3$

7

Running time of insert and search for open addressing

Average case: expected number of probes

sum of the probability of making 1 probe, 2 probes, 3 probes, …

$$E[probes] = 1 + \alpha + \alpha^2 + \alpha^3 + \ldots$$
$$= \sum_{i=0}^{m} \alpha^i$$
$$< \sum_{i=0}^{\infty} \alpha^i$$
$$= \frac{1}{1-\alpha}$$

8

Average number of probes

$$E[probes] = \frac{1}{1-\alpha}$$

α	Average number of searches
0.1	$1/(1-.1) = 1.11$
0.25	$1/(1-.25) = 1.33$
0.5	$1/(1-.5) = 2$
0.75	$1/(1-.75) = 4$
0.9	$1/(1-.9) = 10$
0.95	$1/(1-.95) = 20$
0.99	$1/(1-.99) = 100$

9

How big should a hashtable be?

A good rule of thumb is the hashtable should be around half full

What happens when the hashtable gets full?

Copy: Create a new table and copy the values over

- results in one expensive insert
- simple to implement

Amortized copy: When a certain ratio is hit, grow the table, but copy the entries over a few at a time with every insert

- no single insert is expensive and can guarantee per insert performance
- more complicated to implement

10

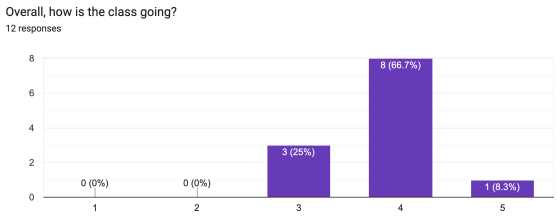
Checkpoint 1

Induction on trees

$$T(n) = T(\sqrt{n}) + c$$

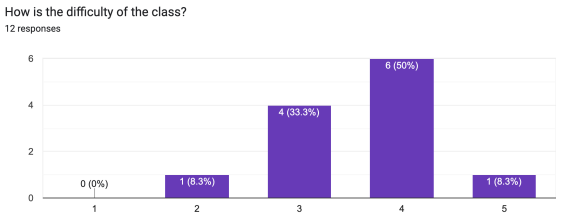
11

Course feedback



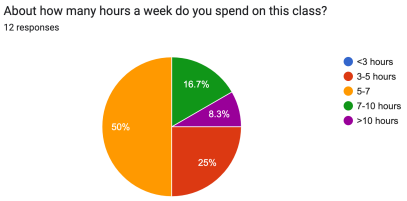
12

Course feedback



13

Course feedback



14

Course feedback

I love proving things and looking at the Math behind the concepts from CS62.

the group assignments

Honestly I just really like the little comics at the start of every homework

15

Course feedback

lectures are wayyy too fast, barely enough time to process things so it feels pointless to take notes; current course content is comprehensive and makes sense but it feels disorganized, like different content stitched together sort of so…

Having more examples, or going through the slides a bit slower

16

Course feedback

The homeworks are a lot of work and the mentors are super helpful but someone's even they don't have the solutions and that wastes hours of our time. I think homeworks can have more straight forward problems that show we understand things rather than problems that we always have to scavenge the internet and bug mentors for understandings.

17

Course feedback

During Class, could we have some more exercises along with the lecture contents?

18

Class overview

Math/Algorithm Tools
- math basics
- big-O (omega and theta)
- recurrences
- amortized analysis
- proofs by induction

Algorithm techniques
- divide and conquer
- greedy
- dynamic programming

19

Class overview

Revisiting data structures
- ArrayList: amortized analysis
- hashtables: big-O analysis
- Redblack trees
- binomial heaps
- disjoint sets

Graphs/Graph Algorithms:
- More details: Dijkstra's, Bellman-Ford, Prim's, Kruskal's
- Topological sort for DAGs
- Floyd-Warshall, Johnsn's (all pairs shortest paths)
- Network flow

20

Class overview

NP completeness

Linear programming

21

https://leetcode.com/problems/largest-number/

179. Largest Number

Medium 6.6K 542

Companies

Given a list of non-negative integers `nums`, arrange them such that they form the largest number and return it.

Since the result may be very large, so you need to return a string instead of an integer.

Example 1:

```
Input: nums = [10,2]
Output: "210"
```

Example 2:

```
Input: nums = [3,30,34,5,9]
Output: "9534330"
```

22

https://leetcode.com/problems/group-anagrams/

Given an array of strings `strs`, group **the anagrams** together. You can return the answer in **any order**.

An **Anagram** is a word or phrase formed by rearranging the letters of a different word or phrase, typically using all the original letters exactly once.

Example 1:

```
Input: strs = ["eat","tea","tan","ate","nat","bat"]
Output: [["bat"],["nat","tan"],["ate","eat","tea"]]
```

Example 2:

```
Input: strs = [""]
Output: [[""]]
```

Example 3:

```
Input: strs = ["a"]
Output: [["a"]]
```

23