

CSC 207 2014S: Extra Session, Week 14

Overview

- Admin
- You ask questions.
- I try to give answers, or at least direct you in the right direction.

Questions and Answers

Should I take the in-class or take-home exam?

I think the in-class is "easier" (in the sense that you don't have to write perfect code, or much code), but you do have to study/think in advance.

The in-class is intended to take less of your time.

Can you do problem 1 on the sample take-home exam?

Sure.

What do we know about skip lists? Expected time to insert something is $O(\log n)$. The time to insert n elements is $n \log n$, which is slightly worse than linear. A not-quite quadratic curve would be good.

For b, Worst case behavior is to insert everything into the same bucket. So $1 + 2 + 3 + \dots$ that's $O(n^2)$. However, the hash table does double in size when it gets pretty full, so we'd have to deal with that issue, too.

Can you do the evil skip list problem (exercise 3)

Basic idea of skip lists is that we have different "height" of nodes, with about $1/(2^h)$ nodes of height h . If this is done right, we can use the one or two nodes at the top height to throw away half the elements, and then again and again at each level.

Insertion:

1. Pick a height to help get the right distribution: 50% probability of height 1, 25% probability of height 2, 12.5% probability of height 3, and so on and so forth.
2. Find where the node belongs
For height = maxheight down to 1
Step along the nodes at height height until we are at the largest value smaller than the goal
3. Insert the node there
4. Make links from all the predecessors identified above

Deletion:

Almost identical. Once again, we gather all the predecessors using the "advance along each level" technique. Then, set `node[level].next` to `node.next[level].next[level]`

Can you do the invariant for computing stamps?

See the (now erased) whiteboard.

How about problem 2 on the take-home?

You should be pretty familiar with the structure of JSON. Yay.

In Sam's parser, Objects are stored as Hashtables and arrays as ArrayLists

Problem: Identify the fields of the object (you know), recursively print at different levels of indentation. Hint: Look at one of the many "dump this tree" methods that Sam wrote.

How about problem 3?

Help on invariants is hard. So let's talk about postconditions. Your goal is that the heap property holds for every index in the array.

Any other comments?

Wow. The take home is more straightforward than I would have thought. Searching our Web site may help for many problems.

Copyright (c) 2013-14 Samuel A. Rebelsky.



This work is licensed under a Creative Commons Attribution 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.