+

David Kauchak
cs312

Review

Midterm

- Will be posted online this afternoon
- You will have 2 hours to take it
  - watch your time!
  - if you get stuck on a problem, move on and come back
- Must take it by Friday at 6pm
- You may use:
  - your book
  - your notes
  - the class notes
  - ONLY these things
- Do NOT discuss it with anyone until after Friday at 6pm

Midterm

- General
  - what is an algorithm
  - algorithm properties
  - pseudocode
  - proving correctness
    - loop invariants
  - run time analysis
  - memory analysis

Midterm

- Big O
  - proving bounds
  - ranking/ordering of functions
- Amortized analysis
- Recurrences
  - solving recurrences
    - substitution method
    - recursion-tree
    - master method

Midterm

- Sorting
  - insertion sort
  - merge sort
    - merge function
  - quick sort
    - partition function
  - bubble sort
  - heap sort

Midterm

- Divide and conquer
  - divide up the data (often in half)
  - recurse
  - possibly do some work to combine the answer
- Calculating order statistics/medians
- Basic data structures
  - set operations
  - array
  - linked lists
  - stacks
  - queues

Midterm

- Heaps
  - binary heaps
  - binomial heaps
- Search trees
  - BSTs
  - B-trees
- Disjoint sets (very briefly)

Midterm

- Other things to know:
  - run-times (you shouldn't have to look all of them up, though I don't expect you to memorize them either)
  - when to use an algorithm
  - proof techniques
    - look again an proofs by induction
      - Make sure to follow the explicit form we covered in class
    - proof by contradiction

+ Proofs

- prove by induction: 1+x^n >= 1+nx for all nonnegative integers n and all x >= −1.

-- base case

-- inductive case

--- inductive hypothesis

--- inductive step to prove

--- proof of inductive step

+ Proofs

- prove by contradiction:

For all integers n, if n2 is odd, then n is odd.

+ Substitution method

$$T(n) = 3T(n/4) + n^2$$

Master method:

$$\text{if } f(n) = O(n^{\log_b a - \varepsilon}) \text{ for } \varepsilon > 0, \text{ then } T(n) = \Theta(n^{\log_b a})$$

$$\text{if } f(n) = \Theta(n^{\log_b a}), \text{ then } T(n) = \Theta(n^{\log_b a} \log n)$$

$$\text{if } f(n) = \Omega(n^{\log_b a + \varepsilon}) \text{ for } \varepsilon > 0 \text{ and } af(n/b) \le cf(n) \text{ for } c < 1$$

$$\text{then } T(n) = \Theta(f(n))$$

+ Substitution method

$$T(n) = 3T(n/4) + n^2$$

Assume $T(k) = O(k^2)$ for all $k < n$

Show that $T(n) = O(n^2)$

Given that $T(n/4) = O((n/4)^2)$, then

$$O(g(n)) = \left\{ f(n): \begin{array}{l} \text{there exists positive constants } c \text{ and } n \text{ such that} \\ 0 \le f(n) \le cg(n) \text{ for all } n \ge n_0 \end{array} \right\}$$

$T(n/4) \le c(n/4)^2$

+ $T(n) = 3T(n/4) + n^2$

To prove that Show that $T(n) = O(n^2)$ we need to identify the appropriate constants:

$$O(g(n)) = \left\{ f(n) : \begin{array}{l} \text{there exists positive constants } c \text{ and } n \text{ such that} \\ 0 \le f(n) \le cg(n) \text{ for all } n \ge n_0 \end{array} \right\}$$

i.e. some constant c such that $T(n) \le cn^2$

$$\begin{aligned} T(n) &= 3T(n/4) + n^2 \\ &\le 3c(n/4)^2 + n^2 \\ &= cn^2 3/16 + n^2 \\ &\le cn^2 \end{aligned}$$

if

$$c \ge \frac{16}{3}$$

+ Changing variables

$$T(n) = 2T(\sqrt{n}) + \log n$$

Guesses?

We can do a variable change: let $m = \log_2 n$ (or $n = 2^m$)

$$T(2^m) = 2T(2^{m/2}) + m$$

Now, let $S(m)=T(2^m)$

$$S(m) = 2S(m/2) + m$$

+ Changing variables

$$S(m) = 2S(m/2) + m$$

Guess? $S(m) = O(m \log m)$

$$T(n) = T(2^m) = S(m) = O(m \log m)$$

substituting m=log n

$$T(n) = O(\log n \log \log n)$$