Group work on divide and conquer
CS24, Dr. Ostheimer

A theorem from the text:

Theorem 1. *Let f be an increasing function that satisfies the recurrence relation*

$$f(n) = af(n/b) + cn^d$$

whenever $n = b^k$, where k is a positive integer, $a \geq 1$, b is an integer greater than 1, and c and d are real numbers with c positive and d nonnegative. Then

- *$f(n)$ is $O(n^d)$ if $a < b^d$,*
- *$f(n)$ is $O(n^d \log n)$ if $a = b^d$, and*
- *$f(n)$ is $O(n^{\log_b a})$ if $a > b^d$.*

(1) Let $t(n) = t(n/2) + 1$. Matching the recurrence relation for t to the theorem above, what are a, b, d?

(2) Which case of the theorem are we in?

(3) What does the theorem predict for the big-O estimate for t?

(4) Use guess and check to find a closed formula for $t(n)$ if $t(1) = 17$.

(5) Find a simple big-O estimate for your closed formula.

(6) Did the theorem give a good prediction?