

Notes on Day 3 slides:

Slides 4-16 are copies of my CSSE 230 slides, so should be entirely review.

Slides 17-24 are from Day 2.

New material starts with Slide 25.

Slide 23: The catch

After Addition first, say,

"The next two bullets are preliminary info that we'll need in order to analyze the runtime of integer addition."

Answers: 2, yes

Slide 25: Multiplication by an Ancient Method

It is correct, because the same numbers get added as in the more "traditional" algorithm.

If we are multiplying n -bit numbers, the number of times through the loop is n .

And each time through the loop requires at most n steps.

And still we have to add up to n numbers of $2n$ bits each.

SO $O(n^2)$

Slide 28: New Multiplication Approach

$T(n) = 4T(n/2) + O(n)$. Solution?

Solution is case 3 of Master Theorem.

$\log_2 4$ is 2, so it is n^2 .

Slide 32: Is this really a lot faster?

Do and solve the recurrence:

$T(n) = 3T(n/2) + \Theta(n)$

In Master theorem, $a = 3$, $b = 2$, $k = 1$.

So we have the third case: $T(N) = \Theta(n^{\log_2 3})$,

Which is approximately $n^{1.59}$.

The file **fib3.py** has the solution, and **fib3-incomplete.py** has it partially completed.