ADA Lecture Notes

Analysis and design of algorithms

These are the notes for analysis and design of algorithms course. The professor says it'll be an interesting course, let's see about that. I am using Obsidian and this is an amazing markdown editor! It has a lot of community plugins. Anyways, study now... xD

Here is a somewhat detailed overview.

1. ADA/Lecture 1 : Introduction to the course and grading.
2. ADA/Lecture 2 : Something more here

Lecture->1

It's like DSA version 2 (in terms of management). Here's the link for previous year: ADA2020, ADA2022. Solutions and questions in this course are made by the instructor and hence making it public is not a good idea. So, these notes with stick around the lectures and maybe sometimes touching things but WILL NOT quote.

Evaluation

- Quizzes : 15% (n-1)
- Homework Assignments (Theory) : 15% (group of two)
- Programming Assignments : 10% (Foobar, No lab hours, Individual)
- Midsem : 30%
- Endsem : 30% Both theory

Multiplying large integers

Input : Two n-digit numbers A and B Output: Product $A \times B$ Primitive Ops: Add/Multiply two single digit integers (recall digital circuits adder)

- Classical pen-paper approach:
  - At max $2n$ operations per partial product, since n, we have $2n^2$
  - Summation of them, $2n^2$
  - Net $4n^2$
- Doing it differently:
  - $\overbrace{56}^{a}\overbrace{78}^{b} \times \overbrace{12}^{c}\overbrace{34}^{d}$
    1. Compute $a.c = 672$
    2. Compute $b.d = 2652$
    3. Compute $(a+b)(c+d) = 6164$
    4. Compute $\boxed{3. - 2. - 1.} = 2840$
    5. Put it all together $6720000 + 2652 + 284000$ (Notice the padding)
    6. Do it all recursively
  - Here's the recursive implementation, where $A = 10^{\frac{n}{2}} \cdot a + b, B = 10^{\frac{n}{2}} \cdot c + d$ and $A \times B = 10^n ac + 10^{\frac{n}{2}}(ad + bc) + bd$
    1. Recursively compute $a.c$
    2. Recursively compute $b.d$
    3. Recursively compute $(a+b) \cdot (c+d)$
    4. Compute $\boxed{3. - 2. - 1.}$ for each call
    5. Pad and add!

Lecture->2