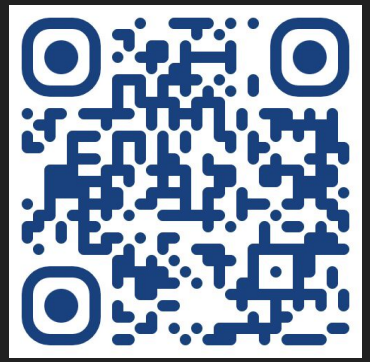
Algorithms and Data Structures



https://n.ethz.ch/~ahmala/and/

Me

- Ahmet Ala
- ahmet.ala@inf.ethz.ch
- discord: ahmet414
- 4th year at ETH, 2nd time AnD TA

About the Course

- 7 Credits, Part of Basisblock 1
- Sorting algorithms, searching algorithms, dynamic programming, graph algorithms…
- Correctness and Run-Time Analysis
- Continues with Algorithms and Probability(second semester Basisblock Course) and Algorithms, Probability and Computing (Major: Theoretical Computer Science)
- But the content appears everywhere e.g. in Theoretische Informatik, Computer Networks, NumCS.

Exercise Session Logistics

- CHN F42
- Monday, 09:15-11.00 (Session)
- Website for slides: https://n.ethz.ch/~ahmala/and/

Homeworks (theoretical part)

- Groups of 2 or 3 (changes in every three weeks)
- New exercise sheet every monday, submitted on Moodle
- Deadline for Exercise Sheet 1 is next Sunday 23:59.
- Only today no peer grading
- 11:00 - 12:00 can be used for peer grading.

In-Class Quizzes

- First time this year
- Starting next week
- 09:15 - 09:30
- Contributes to bonus points
- Online participation possible

Point Distribution

- 12 * {Weekly theoretical exercise sheet (3 points) + peer grading(1 points)} [In Groups]
- 12 * In-class quiz (1 points)
- 5 * {Code Expert programming tasks(biweekly) (6 points)} [Individual]
- min(0.25, 0.25 * n_points / (0.8 * max_points))
- 0.25 bonus for 80% of full possible points

Content of Today

- Induction Proofs
- Comparing functions

When will all dominoes fall?

When will all dominoes fall?

- First domino falls
- Each domino knocks over the next

Induction Proofs

- Base Case(usually n=1): Show, the statement holds for n=1
- Induction Hypothesis(n=k): Assume the statement holds for n=k
- Induction Step(usually n=k+1): Show, the statement also holds for n=k+1
- (usually) By the principle of mathematical induction, this is true for any positive integer n.

Induction Proofs

- Base Case(usually n=1): Show, the statement holds for n=1
- Induction Hypothesis(n=k): Assume the statement holds for n=k
- Induction Step(usually n=k+1): Show, the statement also holds for n=k+1
- (usually) By the principle of mathematical induction, this is true for any positive integer n.

$$\forall P(P(1) \land \forall k(P(k) \implies P(k+1)) \implies \forall n(P(n)))$$

Example

$$\forall P(P(1) \wedge \forall k(P(k) \implies P(k+1)) \implies \forall n(P(n)))$$

Exercise 0.1 *Induction.*

(a) Prove by mathematical induction that for any positive integer n,

$$1 + 2 + \cdots + n = \frac{n(n+1)}{2}.$$

In your solution, you should address the base case, the induction hypothesis and the induction step.

(b) **(This subtask is from August 2019 exam).** Let $T : \mathbb{N} \to \mathbb{R}$ be a function that satisfies the following two conditions:

$$T(n) \geq 4 \cdot T(\tfrac{n}{2}) + 3n \qquad \text{whenever } n \text{ is divisible by 2;}$$
$$T(1) = 4.$$

Prove by mathematical induction that

$$T(n) \geq 6n^2 - 2n$$

holds whenever n is a power of 2, i.e., $n = 2^k$ with $k \in \mathbb{N}_0$. In your solution, you should address the base case, the induction hypothesis and the induction step.

Logarithms

- Quite important for Asymptotic Analysis

$$
\begin{aligned}
\log_a(bc) &= \log_a(b) + \log_a(c) \\
\log_a(b^c) &= c\log_a(b) \\
\log_a(1/b) &= -\log_a(b) \\
\log_a(1) &= 0 \\
\log_a(a) &= 1 \\
\log_a(a^r) &= r \\
\log_{1/a}(b) &= -\log_a(b) \\
\log_a(b)\log_b(c) &= \log_a(c) \\
\log_b(a) &= \frac{1}{\log_a(b)} \\
\log_{a^m}(a^n) &= \frac{n}{m}, \qquad m \neq 0
\end{aligned}
$$

$$
\log_b a = \frac{\log_d a}{\log_d b}
$$

Exercise 0.2 *Comparison of functions part 1.*

Show that

(a) $f(n) := n \log n$ grows asymptotically faster than $g(n) := n$.

(b) $f(n) := n^3$ grows asymptotically faster than $g(n) := 10n^2 + 100n + 1000$.

(c) $f(n) := 3^n$ grows asymptotically faster than $g(n) := 2^n$.

Asymptotic Growth

When we estimate the number of elementary operations executed by algorithms, it is often useful to ignore smaller order terms, and instead focus on the asymptotic growth defined below. We denote by $\mathbb{R}^+$ the set of all (strictly) positive real numbers and by $\mathbb{R}_0^+$ the set of nonnegative real numbers.

Definition 1. Let $f, g : \mathbb{N} \to \mathbb{R}^+$ be two functions. We say that f *grows asymptotically faster than* g if $\lim_{n\to\infty} \frac{g(n)}{f(n)} = 0$.

This definition is also valid for functions defined on $\mathbb{R}^+$ instead of $\mathbb{N}$. In general, $\lim\limits_{n\to\infty} \frac{g(n)}{f(n)}$ is the same as $\lim\limits_{x\to\infty} \frac{g(x)}{f(x)}$ if the second limit exists.

For all the following exercises, you can assume that $n \in \mathbb{N}_{\geq 10}$. We make this assumption so that all functions are well-defined and take values in $\mathbb{R}^+$.

De l'Hopital Rule

Theorem 1 (L'Hôpital's rule). *Assume that functions* $f : \mathbb{R}^+ \to \mathbb{R}^+$ *and* $g : \mathbb{R}^+ \to \mathbb{R}^+$ *are differentiable,* $\lim_{x\to\infty} f(x) = \lim_{x\to\infty} g(x) = \infty$ *and for all* $x \in \mathbb{R}^+$, $g'(x) \neq 0$. *If* $\lim_{x\to\infty} \frac{f'(x)}{g'(x)} = C \in \mathbb{R}_0^+$ *or* $\lim_{x\to\infty} \frac{f'(x)}{g'(x)} = \infty$, *then*

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = \lim_{x\to\infty} \frac{f'(x)}{g'(x)}.$$

Theorem 1 (L'Hôpital's rule). *Assume that functions* $f : \mathbb{R}^+ \to \mathbb{R}^+$ *and* $g : \mathbb{R}^+ \to \mathbb{R}^+$ *are differentiable,* $\lim_{x\to\infty} f(x) = \lim_{x\to\infty} g(x) = \infty$ *and for all* $x \in \mathbb{R}^+$, $g'(x) \neq 0$. *If* $\lim_{x\to\infty} \frac{f'(x)}{g'(x)} = C \in \mathbb{R}_0^+$ *or* $\lim_{x\to\infty} \frac{f'(x)}{g'(x)} = \infty$, *then*

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = \lim_{x\to\infty} \frac{f'(x)}{g'(x)}.$$

Exercise 0.3 *Comparison of functions part 2.*

Show that

(a) $f(n) := n^{1.01}$ grows asymptotically faster than $g(n) := n \ln n$.

(b) $f(n) := e^n$ grows asymptotically faster than $g(n) := n$.

(c) $f(n) := e^n$ grows asymptotically faster than $g(n) := n^2$.

(d)* $f(n) := 1.01^n$ grows asymptotically faster than $g(n) := n^{100}$.

(e) $f(n) := \log_2 n$ grows asymptotically faster than $g(n) := \log_2 \log_2 n$.

(f) $f(n) := 2^{\sqrt{\log_2 n}}$ grows asymptotically faster than $g(n) := \log_2^{100} n$.

(g) $f(n) := n^{0.01}$ grows asymptotically faster than $g(n) := 2^{\sqrt{\log_2 n}}$.

Exercise 0.4 *Simplifying expressions.*

Simplify the following expressions as much as possible without changing their asymptotic growth rates.

Concretely, for each expression $f(n)$ in the following list, find an expression $g(n)$ that is as simple as possible and that satisfies $\lim_{n\to\infty} \frac{f(n)}{g(n)} \in \mathbb{R}^+$.

(a) $f(n) := 5n^3 + 40n^2 + 100$

(b) $f(n) := 5n + \ln n + 2n^3 + \frac{1}{n}$

(c) $f(n) := n \ln n - 2n + 3n^2$

(d) $f(n) := 23n + 4n \log_5 n^6 + 78\sqrt{n} - 9$

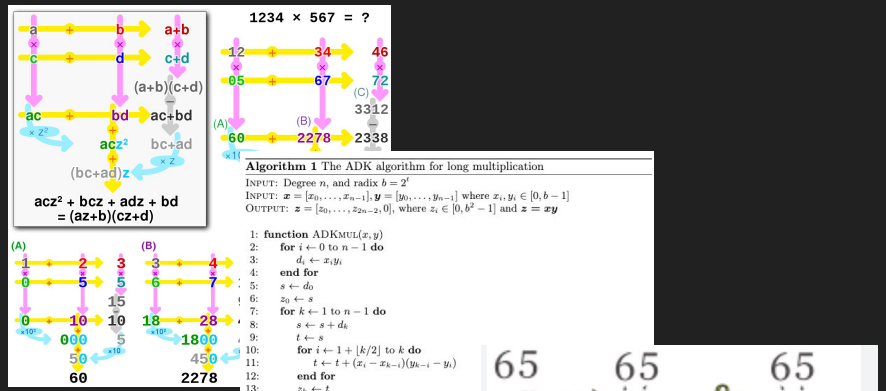
(e) $f(n) := \log_2 \sqrt{n^5} + \sqrt{\log_2 n^5}$

(f)* $f(n) := 2n^3 + (\sqrt[4]{n})^{\log_5 \log_6 n} + (\sqrt[7]{n})^{\log_8 \log_9 n}$

$$1 < log(log(n)) < log(n) < \sqrt{n} < n < n \cdot log(n) < n^2 < 2^n < n!$$

Pasture Break Succeeded

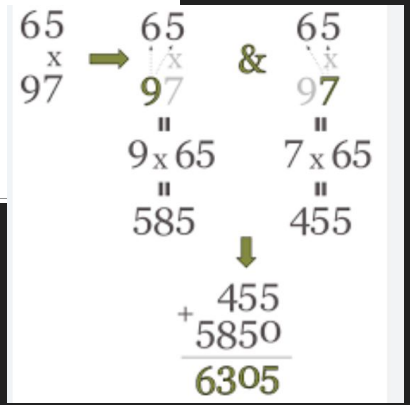
Karatsuba Algorithm(not relevant as well)



Algorithm 1 The ADK algorithm for long multiplication

INPUT: Degree n, and radix $b = 2^t$
INPUT: $\boldsymbol{x} = [x_0, \ldots, x_{n-1}], \boldsymbol{y} = [y_0, \ldots, y_{n-1}]$ where $x_i, y_i \in [0, b-1]$
OUTPUT: $\boldsymbol{z} = [z_0, \ldots, z_{2n-2}, 0]$, where $z_i \in [0, b^2 - 1]$ and $\boldsymbol{z} = \boldsymbol{xy}$

```
1: function ADKMUL(x, y)
2:     for i ← 0 to n − 1 do
3:         d_i ← x_i y_i
4:     end for
5:     s ← d_0
6:     z_0 ← s
7:     for k ← 1 to n − 1 do
8:         s ← s + d_k
9:         t ← s
10:        for i ← 1 + ⌊k/2⌋ to k do
11:            t ← t + (x_i − x_{k−i})(y_{k−i} − y_i)
12:        end for
13:        z_k ← t
14:    end for
15:    for k ← n to 2n − 2 do
16:        s ← s − d_{k−n}
17:        t ← s
18:        for i ← 1 + ⌊k/2⌋ to n − 1 do
19:            t ← t + (x_i − x_{k−i})(y_{k−i} − y_i)
20:        end for
21:        z_k ← t
22:    end for
23:    return z
24: end function
```



Groups