Week 2

Big-O Notation

Thorben Klabunde

www.th-kl.ch

29.09.2025

Agenda

Mini-Quiz

Quiz 1

1. **Cl.:** For $n \in \mathbb{N}$ let $f(n) = n^2 + 1001n + n^3$ and $g(n) = 10n^3$. Then $f(n)$ has the same asymptotic growth rate as $g(n)$ (meaning that $\lim_{n\to\infty} \frac{f(n)}{g(n)} \in \mathbb{R}_+$).

   Answer: True, both are polynomials of degree 3

2. **Cl.:** $n^4 \leq O(\frac{n^4}{\log n})$

   Answer: False, since $\lim\limits_{n\to\infty} \frac{n^4}{\frac{n^4}{\log n}} = \lim\limits_{n\to\infty} \frac{n^4 \log n}{n^4} = \infty$

3. **Cl.:** $e^{3\ln(n)} \leq O(n^2)$

   Answer: False, since $e^{3\ln(n)} = n^3 \not\leq O(n^2)$

Quiz 1

4. **Cl.:** Let $f(n) = 6n^2 + 5n + 10$ (for $n \in \mathbb{N}$). For each of the following definitions of $g(n)$, is $g(n) \leq O(f(n))$?

True	False	
✓		$g(n) = 123456789n - 200\sqrt{n}$
	✓	$g(n) = 0.01n^2 \log(n)$
	✓	$g(n) = 10n^3 + 5n + 1000$

Quiz 1

5. **Cl.:** Let $f : \mathbb{N} \to \mathbb{R}_+$ be some function for which we would like to prove $f(n) \geq n^2$ for every $n \geq 1$. Assume that you have proven that:
   - $f(2) \geq 2^2$
   - if $f(k) \geq k^2$ holds for an arbitrary positive integer k, then $f(k+1) \geq (k+1)^2$ holds.

   Then, $f(n) \geq n^2$ holds for all positive integers $n \geq 1$.

   Answer: False, incorrect choice of B.C., $n = 1$ is never proved.

Assignment

Exercise 1.1 *Sum of Cubes* **(1 point).**

Prove by mathematical induction that for every positive integer n,

$$1^3 + 2^3 + \cdots + n^3 = \frac{n^2(n+1)^2}{4}.$$

cl. $\forall n \in \mathbb{N}^+$ it holds: $\sum_{i=1}^{n} i^3 = \frac{n^2(n+1)^2}{4}$

pr. We proceed by induction on n.

B.C.: Let $n=1$. Then: $\sum_{i=1}^{n} i^3 = 1^3 = \frac{1(1+1)^2}{4}$. The B.C. holds.

I.H.: Assume for some $n \in \mathbb{N}^+$ the claim holds.

I.S.: Then for $n+1$ we have:

$$n^2+4n+4 = (n+2)(n+2)$$

$$\sum_{i=1}^{n+1} i^3 = \left(\sum_{i=1}^{n} i^3\right) + (n+1)^3 \overset{\text{I.H.}}{=} \frac{n^2(n+1)^2}{4} + (n+1)^3 = \frac{n^2(n+1)^2 + 4(n+1)^3}{4} = \frac{(n+1)^2(n^2+4(n+1))}{4} = \frac{(n+1)^2(n+2)^2}{4}$$

By the principle of math. induction we have proven the claim.

Ex. 1.2 - Remarks on Proof Technique

Key Elements of an Inductive Proof

When writing the inductive step, you must clearly distinguish between:

- **The statement to be proved.**
  - Here: Show that $\sum_{i=1}^{n+1} \frac{1}{\sqrt{i}} \leq \sqrt{n+1}$ holds (**it doesn't**, don't try to show it ;))
- **The allowed assumptions.**
  - Here, the I.H.: Assume $\sum_{i=1}^{n} \frac{1}{\sqrt{i}} \leq \sqrt{n}$ is true for some $n \in \mathbb{N}$.

Ex. 1.2 - Remarks on Proof Technique

Warning: Correct proofs must build from valid assumptions towards the desired conclusion.

- **Correct Logic:** Start with the I.H. (a true statement, A) and show through a series of valid steps that it implies your goal (statement B). This proves B.
- **Fallacies:**
  - Starting with your goal (B) and showing it implies a known statement A. This only proves the implication $B \implies A$, but not B itself. This would require **equivalences** ($\iff$).
  - Not checking **both implications** for equivalences ($\iff$)
  - **Only proving** the **implication** $A \implies B$ but not the statement A, does not allow us to conclude B.

Always make sure you proved all statements that you cannot assume! Writing out your proofs carefully, step-by-step with explicit explanations ensures this.

Ex. 1.2)

Exercise 1.2 *Sum of reciprocals of roots* **(1 point).**

Consider the following claim:

$$\frac{1}{\sqrt{1}} + \frac{1}{\sqrt{2}} + \cdots + \frac{1}{\sqrt{n}} \leq \sqrt{n}.$$

A student provides the following induction proof. Is it correct? If not, explain where the mistake is.

Base case: $n = 1$,

$$\frac{1}{\sqrt{1}} \leq 1, \text{ which is true.}$$

Induction hypothesis: Assume the claim holds for $n = k$, i.e.

$$\frac{1}{\sqrt{1}} + \cdots + \frac{1}{\sqrt{k}} \leq \sqrt{k}.$$

Induction step: Then, starting from the claim we need to prove for $n = k + 1$ and using logical equivalences:

$$\frac{1}{\sqrt{1}} + \cdots + \frac{1}{\sqrt{k}} + \frac{1}{\sqrt{k+1}} \leq \sqrt{k+1} \iff \frac{1}{\sqrt{1}} + \cdots + \frac{1}{\sqrt{k}} \leq \sqrt{k+1} - \frac{1}{\sqrt{k+1}}$$

$$\iff \frac{1}{\sqrt{1}} + \cdots + \frac{1}{\sqrt{k}} \leq \frac{k+1}{\sqrt{k+1}} - \frac{1}{\sqrt{k+1}}$$

$$\iff \frac{1}{\sqrt{1}} + \cdots + \frac{1}{\sqrt{k}} \leq \frac{k}{\sqrt{k+1}} \leq \frac{k}{\sqrt{k}} \leq \sqrt{k},$$

which is true, therefore the claim holds by the principle of mathematical induction.

With the points from the prev. slide in mind, notice:

We are allowed to assume (____): $\frac{1}{\sqrt{1}} + \ldots + \frac{1}{\sqrt{k}} \leq \sqrt{k}$

and need to derive (____).

Problem:

The proof does not follow this pattern!

Notice: Stmt. (____) cannot be assumed and

$$\frac{1}{\sqrt{1}} + \ldots + \frac{1}{\sqrt{k}} \leq \frac{k}{\sqrt{k}} = \sqrt{k}$$

$$\not\Rightarrow \frac{1}{\sqrt{1}} + \ldots + \frac{1}{\sqrt{k}} \leq \frac{k}{\sqrt{k+1}}.$$

Hence, the chain of implications from (____) to (____) is broken.

We only prove $P(k+1) \Rightarrow P(k)$ but not $P(k) \Rightarrow P(k+1)$.

Be careful with equivalences & "backward proofs".

Exercise 1.4 *Proving Inequalities.*

(a) Prove the following inequality by mathematical induction

$$\frac{1}{2}\cdot\frac{3}{4}\cdot\frac{5}{6}\cdot\ldots\cdot\frac{2n-1}{2n}\le\frac{1}{\sqrt{3n+1}},\quad n\ge 1.$$

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

cl. For $n\geq 1$, $\frac{1}{2}\cdot\frac{3}{4}\cdot\frac{5}{6}\cdot\ldots\cdot\frac{2n-1}{2n}\leq\frac{1}{\sqrt{3n+1}}$.

pr. We proceed by induction on n.

B.C.: Let $n=1$. Then notice: $\frac{1}{2}\leq\frac{1}{\sqrt{4}}$ holds true.

I.H.: Assume that for some $n\geq 1$ the claim holds.

I.S.: Then for $n+1$, we have:

$$\frac{1}{2}\cdot\frac{3}{4}\cdot\ldots\cdot\frac{2(n+1)-1}{2(n+1)}=\prod_{i=1}^{n+1}\frac{2i-1}{2i}=\left(\prod_{i=1}^{n}\frac{2i-1}{2i}\right)\cdot\frac{2(n+1)-1}{2(n+1)}\overset{\text{I.H.}}{\leq}\frac{1}{\sqrt{3n+1}}\cdot\frac{2(n+1)-1}{2(n+1)}\overset{(1)}{\leq}\frac{1}{\sqrt{3(n+1)+1}}$$

Proof on the following page

Exercise 1.4 *Proving Inequalities.*

(a) Prove the following inequality by mathematical induction

$$\frac{1}{2} \cdot \frac{3}{4} \cdot \frac{5}{6} \cdot \ldots \cdot \frac{2n-1}{2n} \leq \frac{1}{\sqrt{3n+1}}, \quad n \geq 1.$$

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

Proof of (1): $\frac{1}{\sqrt{3n+1}} \cdot \frac{2(n+1)-1}{2(n+1)} \leq \frac{1}{\sqrt{3(n+1)+1}} = \frac{1}{\sqrt{3n+4}}$

$\Leftrightarrow \frac{1}{3n+1} \cdot \left(\frac{2(n+1)-1}{2(n+1)}\right)^2 \leq \frac{1}{3n+4}$ (both sides strictly pos.)

$\Leftrightarrow (3n+1) \cdot (2n+2)^2 \geq (3n+4)(2n+1)^2$

$\Leftrightarrow (3n+1) \cdot (4n^2+8n+4) \geq (3n+4)(4n^2+4n+1)$

$\Leftrightarrow 12n^3+28n^3+20n+4 \geq 12n^3+28n^2+19n+4$

$\Leftrightarrow n \geq 0$

Ex. 1.3b)

(b) $f(m) = \log(m^3)$ grows asymptotically slower than $g(m) = (\log m)^3$.

(b) $$\lim_{m\to\infty} \frac{f(m)}{g(m)} = \lim_{m\to\infty} \frac{\log(m^3)}{\log(m)^3} = \lim_{m\to\infty} \frac{3\log(m)}{\log(m)^3} = \lim_{m\to\infty} \frac{3}{\log(m)^2} = 0$$

$\Rightarrow$ $f(m)$ grows asymptotically slower than $g(m)$.

(c) $f(m) = e^{2m}$ grows asymptotically slower than $g(m) = 2^{3m}$.

Hint: *Recall that for all* $n, m \in \mathbb{N}$, *we have* $n^m = e^{m \ln n}$.

(c) $\lim_{m \to \infty} \frac{f(m)}{g(m)} = \lim_{m \to \infty} \frac{e^{2m}}{2^{3m}} = \lim_{m \to \infty} \Big(\underbrace{\frac{e^2}{2^3}}_{\in (0,1)}\Big)^m = 0$

$\Rightarrow$ $f(m)$ grows asymptotically slower than $g(m)$.

Ex. 1.3d)

(d)* If $f(m)$ grows asymptotically slower than $g(m)$, then $\log(f(m))$ grows asymptotically slower than $\log(g(m))$.

(d) False. Consider $f(n) = n$ and $g(n) = n^2$.

Then: $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim \frac{n}{n^2} = 0$ and $f(n)$ grows asymptotically slower than $g(n)$.

But: $\lim_{n\to\infty} \frac{\log(f(n))}{\log(g(n))} = \lim_{n\to\infty} \frac{\log(n)}{\log(n^2)} = \lim_{n\to\infty} \frac{\log(n)}{2\log(n)} = \frac{1}{2} \neq 0$,

wherefore $\log(f(n))$ & $\log(g(n))$ have the same asymptotic growth rate.

Ex. 1.3e)

(e)* $f(m) = \ln(\sqrt{\ln(m)})$ grows asymptotically slower than $g(m) = \sqrt{\ln(\sqrt{m})}$.

***Hint:** You can use L'Hôpital's rule from sheet 0.*

(e) $$\lim_{m\to\infty} \frac{\ln(\sqrt{\ln(m)})}{\sqrt{\ln(\sqrt{m})}} \overset{\text{Hôp.}}{=} \lim_{m\to\infty} \frac{\dfrac{\frac{1}{2}\ln(m)^{-\frac{1}{2}} \cdot \frac{1}{m}}{\sqrt{\ln(m)}}}{\frac{1}{2}\ln(\sqrt{m})^{-\frac{1}{2}} \cdot \frac{1}{\sqrt{m}} \cdot \frac{1}{2} \cdot \frac{1}{\sqrt{m}}}$$

"$\frac{\infty}{\infty}$" case and no div by 0

$$= \lim_{m\to\infty} \frac{\frac{1}{2m\cdot\ln(m)}}{\frac{1}{4m\sqrt{\ln(\sqrt{m})}}} = \lim_{m\to\infty} \frac{2\sqrt{\ln(\sqrt{m})}}{\ln(m)} = \lim_{m\to\infty} \frac{2\sqrt{\frac{1}{2}\ln(m)}}{\ln(m)}$$

$$= \lim_{m\to\infty} \frac{2\cdot\frac{1}{4}\cdot\sqrt{\ln(m)}}{\ln(m)} = \lim_{m\to\infty} \frac{1}{2\sqrt{\ln(m)}} = 0$$

Theory Recap

The Challenge: How to Measure Efficiency?

Why can't we just time our code with a stopwatch?

- The **execution time** of an algorithm **depends on the specific hardware**.
  - CPU speed and microarchitecture
  - Available memory (RAM)
  - ...
- It also depends on the **software environment**.
  - Programming language and compiler
  - Operating System

$\Longrightarrow$ We need a **common frame for comparisons** that is **independent** of these factors.

Solution Part 1: A Universal Model

We abstract away from the specific hardware by creating a simplified model.

The Unit-Cost RAM Model (Random Access Machine)

Instead of measuring seconds, we count the number of **basic operations** an algorithm performs.

- A **basic operation** is an instruction that takes a **constant** amount of time.
- Examples:
  - Arithmetic $(+, -, *, /)$
  - Comparisons $(<, >, ==)$
  - Memory access (assignments)

This gives us a runtime measured in number of operations, making our analysis **machine-independent**.

Solution Part 2: Asymptotic Analysis

Given an input instance I (a bit-string), we measure the number of operations as a function of the length n of I.

- **Problem:** Which input length to analyze?
- Instead of assuming a **specific input length**, analyze the **growth rate** of the runtime.

Asymptotic Analysis

We analyze the growth rate of the runtime as the input size n approaches infinity ($n \to \infty$).

Putting It All Together: Big-O Notation

Big-O notation combines these two ideas. It describes the **asymptotic upper bound** on the number of operations.

Big-O Notation

A function $f(n)$ is in the set $O(g(n))$, written:

$$f(n) \in O(g(n)) \quad \text{resp. in this course using the notation } f(n) \leq O(g(n)),$$

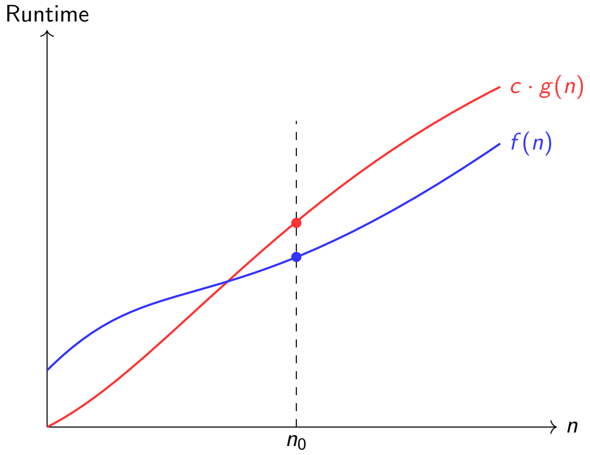
if there exist positive constants c and n_0 such that for all $n \geq n_0$:

$$0 \leq f(n) \leq c \cdot g(n)$$

In plain English: For all large inputs ($n \geq n_0$), $f(n)$ is "less than or equal to" $c \cdot g(n)$ (it is *upper bounded* by g(n)), for some constant factor $c > 0$.

Visualizing Big-O

The function $f(n)$ is our algorithm's runtime. After the point n_0, it is always below the curve of $c \cdot g(n)$.



Caution: Intuition Can Be Misleading

Asymptotic behavior only dominates for **large** values of n.

Example: Which algorithm is "better"?

- Algorithm A has a runtime of $T_A(n) = n^{1000}$.
- Algorithm B has a runtime of $T_B(n) = 1.01^n$.

Asymptotically, Algorithm A is better: $O(n^{1000})$ grows slower than $O(1.01^n)$.

Advice: Trust the formal definitions and analyze the structure (substitute values for variables that contain the critical information), don't try to mentally plot the functions:

- A is a polynomial n^k, $k \in \mathbb{N}$, whereas B is an exponential c^n with $c > 1$. $\implies$ B dominates A!

Limitations of Big-O Notation

While a sensible measure in many cases, Big-O is **not the whole story**.

- **Constant factors do matter** in practice. An algorithm running in $2n$ steps is better than one running in $1000n$ steps, even though both are $O(n)$.
- The **asymptotic view isn't always relevant**. If your application's input size is always small (e.g., $n < 100$), the asymptotically "slower" algorithm might be faster in practice.
- Big-O describes the **worst-case** scenario. Sometimes, the average-case or best-case performance is very different.

Summary: Key Takeaways

1. We need to analyze algorithms in a way that is **independent of hardware** and specific inputs.
2. We do this by **counting basic operations** as a function of input size n.
3. We analyze the **asymptotic growth rate** ($n \to \infty$) to understand how the algorithm scales.
4. **Big-O notation** provides a formal language for the **asymptotic upper bound**, ignoring constant factors.
5. It's a powerful theoretical tool, but always remember its **practical limitations**.

Additional Practice

Exercise 1.2 *Sums of powers of integers.*

(a) Show that, for all $n \in \mathbb{N}_0$, we have $\sum_{i=1}^{n} i^3 \leq n^4$.

(b) Show that for all $n \in \mathbb{N}_0$, we have $\sum_{i=1}^{n} i^3 \geq \frac{1}{2^4} \cdot n^4$.

Hint: *Consider the second half of the sum, i.e.,* $\sum_{i=\lceil \frac{n}{2} \rceil}^{n} i^3$. *How many terms are there in this sum? How small can they be?*

(b) Let $n \in \mathbb{N}_0$ be arbitrary. Then:

$$\sum_{i=1}^{n} i^3 = \underbrace{\sum_{i=1}^{\lceil \frac{n}{2} \rceil - 1} i^3}_{\geq 0} + \sum_{i=\lceil \frac{n}{2} \rceil}^{n} i^3 \geq \sum_{i=\lceil \frac{n}{2} \rceil}^{n} i^3 \geq \sum_{i=\lceil \frac{n}{2} \rceil}^{n} \left(\frac{n}{2}\right)^3 \geq \underbrace{\left(n - \left(\lceil \tfrac{n}{2} \rceil - 1\right)\right)}_{\geq \frac{n}{2}} \left(\frac{n}{2}\right)^3 \geq \left(\frac{n}{2}\right)^4$$

$\left(\frac{n}{2} \leq \lceil \frac{n}{2} \rceil\right)$

Subtract number of deleted term

$$\lceil \tfrac{n}{2} \rceil^3 + \left(\lceil \tfrac{n}{2} \rceil + 1\right)^3 + \dots + (n)^3 \geq \left(\tfrac{n}{2}\right)^3 + \left(\tfrac{n}{2}\right)^3 + \dots + \left(\tfrac{n}{2}\right)^3$$

Each term is '≤'

2024 - Ex. 1.2 ctd.

Together, these two inequalities show that $C_1 \cdot n^4 \leq \sum_{i=1}^{n} i^3 \leq C_2 \cdot n^4$, where $C_1 = \frac{1}{2^4}$ and $C_2 = 1$ are two constants independent of n. Hence, when n is large, $\sum_{i=1}^{n} i^3$ behaves "almost like n^4" up to a constant factor.

(c)* Show that parts (a) and (b) generalise to an arbitrary $k \geq 4$, i.e., show that $\sum_{i=1}^{n} i^k \leq n^{k+1}$ and that $\sum_{i=1}^{n} i^k \geq \frac{1}{2^{k+1}} \cdot n^{k+1}$ holds for any $n \in \mathbb{N}_0$.

Let $k \geq 4$ and $n \in \mathbb{N}^0$ be arbitrary. Then:

$$\sum_{i=1}^{n} i^k \leq \sum_{i=1}^{n} n^k = n \cdot n^k = n^{k+1}$$

$$\sum_{i=1}^{n} i^k = \underbrace{\sum_{i=1}^{\lceil \frac{n}{2} \rceil - 1} i^k}_{\geq 0} + \sum_{i=\lceil \frac{n}{2} \rceil}^{n} i^k \geq \sum_{i=\lceil \frac{n}{2} \rceil}^{n} i^k \geq \sum_{i=\lceil \frac{n}{2} \rceil}^{n} \left(\frac{n}{2}\right)^k \geq \underbrace{\left(n - \left(\lceil \tfrac{n}{2} \rceil - 1\right)\right)}_{\geq \frac{n}{2}} \left(\frac{n}{2}\right)^k \geq \left(\frac{n}{2}\right)^{k+1}$$

(i^k is str. mon. incr. for $k \geq 4$ & $n \in \mathbb{N}^0$)

Peer Grading

Peer-Grading Exercise

This week's peer-grading exercise is **Exercise 1.1.**

Each group grades the group below in the table I sent you (resp. the last one grades the first one). Please send the other group your solution. If you don't get their solution, please contact me so I can send it to you.