

# Algorithmen & Datenstrukturen

Woche 1

---

Marius Tomek, Nicolas Wehrli, Tim Rieder

25. September 2022

ETH Zürich

## Kontakt

- In Übungsstunde
- Per Mail an [nwehrl@student.ethz.ch](mailto:nwehrl@student.ethz.ch)
- Discord: [@Nicolas \[A&D\]](#)
- Interesse an Whatsapp Chat?

## Was ihr sonst noch wissen müsst

- Prüft die Webseite regelmässig:  
<https://cadmo.ethz.ch/education/lectures/HS22/DA>
- Dort findet ihr zum Beispiel: Am 29.9. und am 6.10. findet am Nachmittag EProg anstelle von A&D statt.

## Aufgaben (zählt 3x)

- Beispiel aus letztem Jahr: Stern-Aufgaben & Bonus-Aufgaben
- In Zweiergruppen lösen (wechseln alle 3 Wochen)
- Erhalt nach der Stunde, Abgabe vor nächster Stunde
- Abgaben entweder über Polybox oder auf Papier

## PeerGrading (zählt 1x)

- Grundsätzlich: 11:15 - 12:00. Meist früher fertig
- Aber: Abgabefrist ist Montag, 23:59 falls nötig.

## Programmieraufgaben (zählt 4x)

*80% der Bonuspunkte → +0.25 bei der Endnote*

	HS19	HS20	HS21
Diskrete Mathematik	3.88	3.72	3.94
Einführung in die Programmierung	4.31	4.24	3.93
Lineare Algebra	4.27	4.09	4.28
<b>Algorithmen &amp; Datenstrukturen</b>	<b>4.16</b>	<b>4.19</b>	<b>4.13</b>

# Ablauf Übungsstunden

- Recap (Vorlesung / Übungen)
- Übungen (vor-)lösen (letzte Woche / neue Übungen)
- Ausblick
- Peer-Grading

*Heute: Induktion, Asymptotisches Wachstum, Exercise 0, früher gehen, da kein Peer-Grading stattfindet :)*

Mittels Induktion beweisen wir Aussagen der Art:

«Zeige, dass ... für alle natürlichen Zahlen gilt »

oder ähnlichen Fragestellungen (für alle  $n = 2^k$ , für alle  $n \geq 10$ )

---

$$\sum_{k=1}^n k = \frac{n(n+1)}{2}.$$

## **Base Case.**

Zeige, dass die Formel für  $n = 1$  gilt.

## **Induction Hypothesis.**

Nehme an, die Formel gilt für gewisse  $k$ .

## **Inductive Step.**

Zeige, dass die Formel auch für  $k + 1$  gilt, falls sie für  $k$  gilt (I.H.).

⇒ Nach dem Prinzip der mathematischen Induktion gilt die Aussage für alle positiven Integer  $n$ .

# Asymptotisches Wachstum

*In A&D sind wir oft daran interessiert, wie sich Algorithmen für verschieden grosse und für sehr grosse Eingaben verhalten.*

---

Beispiel: **Asymptotisches Wachstum Geogebra**

Beispiel: Zeige, dass  $n \cdot \log n$  asymptotisch schneller wächst als  $n$ .

$$\lim_{n \rightarrow \infty} \frac{n}{n \log n} = \lim_{n \rightarrow \infty} \frac{1}{\log n} \rightarrow 0.$$

Beispiel: Fibonacci-Folge kann man in exponentieller Zeit ( $\mathcal{O}(2^n)$ ) oder aber in linearer Zeit ( $\mathcal{O}(n)$ ) berechnen.