

Übungsblatt 4 zur Vorlesung Algorithmen und Datenstrukturen (Sommer 2026)

Abgabe: Bis 2026-05-15 18:00, on ILIAS.

1. Aufgabe (Stirling's bound)

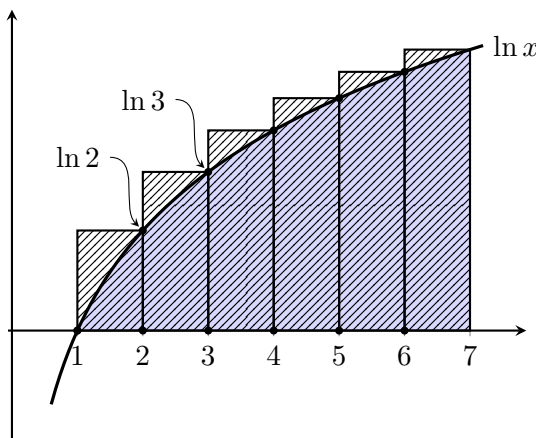
10 + 20 + 20 Punkte

- a) Beweisen Sie: $\forall n \in \mathbb{N}_{\geq 0} : \ln(n!) \leq n \ln n$.

Tipp: $\ln(xy) = \ln(x) + \ln(y)$

- b) Beweisen Sie: $\forall n \in \mathbb{N}_{\geq 2} : \ln(n!) \geq (n-1) \ln(n-1) - n$.

Tipp: Verwenden Sie die Tatsache, dass eine Summe $f(1) + \dots + f(n)$ durch $\int_2^n f(x) dx$ nach unten beschränkt werden kann für konkave Funktionen f .



- c) Geben Sie eine \sim -Asymptotik für $\ln(n!)$ an.

Tipp: Verwenden Sie a) und b), sowie

$$\ln(n-1) = \ln\left(n \cdot \frac{n-1}{n}\right) = \ln(n) + \ln\left(1 - \frac{1}{n}\right) = \ln(n) \pm O\left(\frac{1}{n}\right)$$

(Der letzte Schritt folgt aus der Taylor-Reihenentwicklung von $\ln(x)$ um $x = 1$; aber das müssen Sie für ADS nicht unbedingt wissen 😊)

2. Aufgabe (Monty Hall)

20 Punkte

Sie sind Kandidat in einer Spielshow. Der Moderator, Monty, zeigt Ihnen drei geschlossene Türen.

Hinter einer Tür befindet sich ein fabrikneues Auto.

Hinter den anderen beiden Türen stehen Ziegen.


Das Spiel läuft in 3 Schritten ab:

1. Sie wählen eine Tür aus (sagen wir Tür #1). Sie hoffen, dass das Auto dort ist, aber Sie öffnen sie noch nicht.
2. Monty, der genau weiß, was sich hinter jeder Tür befindet, öffnet eine der anderen Türen (z. B. Tür #3) und bringt eine Ziege zum Vorschein.
3. Monty sieht Sie an und fragt: „Möchten Sie bei Tür #1 bleiben oder möchten Sie zu Tür #2 wechseln?“

Analysieren Sie Ihre Gewinnwahrscheinlichkeit für das Auto bei den zwei möglichen Strategien: „Bleiben“ und „Wechseln“.

3. Aufgabe (Percolation)





50 Punkte

Lösen Sie marburg.kilonova.ro/problems/2 (Percolation) .

ILIAS-Abgabe

Beschreiben Sie in Ihrer ILIAS-Abgabe Ihren verwendeten Algorithmus entsprechend des Templates aus der Vorlesung. Der Pseudocode-Teil kann hier eine informelle Zusammenfassung Ihres eingereichten Codes auf Kilonova sein.

Template für Entwurf von Algorithmen

1.  **Algorithmische Idee**
Kurze abstrakte Idee, die dem Algorithmus zu Grunde liegt (Freitext)
(ein Experte könnte den Rest von hier ergänzen)
2.  **Pseudocode**
strukturierte Beschreibung der Methode inklusive Rand- und Basisfälle
eindeutig, präzise; nah an echtem Code
3.  **Korrektheitsbeweis**
Argument, warum der Algorithmus das korrekte Ergebnis liefert
oft per Induktion unter Verwendung von Invarianten
4.  **Analyse**
Analyse der Kosten des Algorithmus
i.d.R. Θ -Klasse für den Worst Case;
wo interessant auch Speicherbedarf