

CHRISTIAN-ALBRECHTS-UNIVERSITÄT ZU KIEL

Institut für Informatik, Arbeitsgruppe Algorithmen und Komplexität

Prof. Dr. K. Jansen

Übungsmaterial zur Klausurvorbereitung

CAU-Siegel
(rundes
Universitäts-
siegel)

Probeklausur 1

Analyse von Algorithmen und Komplexität, SS 2026

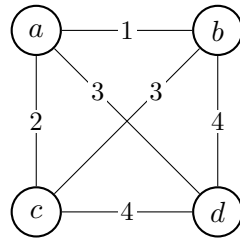
Name: _____ Matrikel-Nr.: _____

Hinweise:

- Bearbeiten Sie *alle* Aufgaben.
- Sie haben 10 Minuten Einlesezeit. Danach haben Sie **120 Minuten** Zeit für die Klausurbearbeitung.
- Erlaubte Hilfsmittel: Ein doppelseitig handschriftlich beschriebenes Blatt Papier DIN A4, (bunte) Stifte (kein Rot, kein Grün, kein Bleistift).
- Die Klausur besteht aus **5 Aufgaben**, die alle **10 Punkte** wert sind.
- Zum Bestehen müssen mindestens **18 Punkte** erreicht werden.
- Aufgabe 5 gibt zusätzlich **10 Bonuspunkte**. Insgesamt können also **50 Punkte plus 10 Bonuspunkte** erreicht werden.

Aufgabe 1 ANWENDUNG – CHRISTOFIDES’ ALGORITHMUS (6 + 4 Punkte)

Gegeben sei folgender vollständiger, symmetrischer Graph $G = (V, E)$ auf den Knoten $\{a, b, c, d\}$. Die Dreiecksungleichung gilt für jede Kante.



- a) Wenden Sie den Algorithmus von Christofides (Δ TSP2) auf den Graphen G mit Startknoten a an. Geben Sie dabei alle Graphen an, die in Zwischenschritten entstehen. Erwähnen Sie ebenfalls Schritte, die bei dieser Anwendung zu keiner Änderung führen, die der Algorithmus aber überprüfen muss. Geben Sie außerdem die Länge der berechneten Rundreise an.
- b) Was berechnet der Algorithmus von Christofides? Für die Korrektheit werden zwei zusätzliche Eigenschaften für die Eingabe des Travelling-Salesperson-Problems benötigt. Nennen Sie die beiden Eigenschaften und geben Sie jeweils eine Eingabe an, welche die Eigenschaft *nicht* erfüllt. Geben Sie auch die Güte des Algorithmus an.

Aufgabe 2 VORLESUNGSBEWEIS – DEFINITION NP (2 + 8 Punkte)

Beweisen Sie folgende Aussage:

$$\{L \subseteq \Sigma^* \mid \exists \text{ polynomieller Verifizierer für } L\} \subseteq \\ \{L \subseteq \Sigma^* \mid \exists \text{ NDTM } M \text{ über } \Gamma \supset \Sigma \text{ und ein Polynom } T, \\ \text{sodass } M \text{ die Sprache } L \text{ in nicht-deterministischer Zeit } T \text{ akzeptiert}\}$$

Ausformuliert bedeutet das: Sei $L \subseteq \Sigma^*$ eine Sprache, für die ein polynomieller Verifizierer existiert. Nun müssen Sie folgende Aussage beweisen: Es existiert eine NDTM M über einem Alphabet $\Gamma \supset \Sigma$ und ein Polynom T , sodass L von M in nicht-deterministischer Zeit T entschieden wird.

- a) Geben Sie dazu zunächst für die Sprache L , die durch folgenden polynomiellen Verifizierer V akzeptiert wird, eine NDTM M an, die in analoger Weise arbeitet. Der Verifizierer V erhält als Eingabe einen ungerichteten Graphen $G = (V, E)$ und eine Zahl $k \in \mathbb{N}_0$ sowie als Zertifikat eine Knotenmenge $C \subseteq V$.

```
V(G = (V, E), k; Zertifikat C ⊆ V)
1. Falls |C| > k, verwerfe.
2. Für alle {u, v} ∈ E: Falls u ∉ C und v ∉ C, verwerfe.
3. Akzeptiere.
```

- b) Führen Sie nun einen allgemeingültigen Beweis.

Aufgabe 3 NP-VOLLSTÄNDIGKEIT

(10 Punkte)

Wir betrachten folgendes Problem:

Problem: k -CLIQUEMEMBER

Eingabe: Ein ungerichteter Graph $G = (V, E)$, ein Knoten $v \in V$ sowie eine Zahl $k \in \mathbb{N}_{\geq 1}$.

Entscheide: Existiert eine Clique $C \subseteq V$ mit $|C| \geq k$ und $v \in C$?

Zeigen Sie, dass k -CLIQUEMEMBER NP-vollständig ist.

Hinweis: CLIQUE (Gegeben (G, k) : existiert eine Clique $C \subseteq V$ mit $|C| \geq k$?) ist bereits als NP-vollständig bekannt.

Problem: VERTEXCOVER**Eingabe:** Ein ungerichteter Graph $G = (V, E)$ und eine Zahl k .**Entscheide:** Gibt es $S \subseteq V$ mit $|S| \leq k$, sodass $v \in S$ oder $u \in S$ für jede Kante $\{v, u\} \in E$?Betrachten Sie folgende Reduktion von CLIQUE auf VERTEXCOVER. Für eine CLIQUE-Instanz $(G = (V, E), k)$ bilde die Instanz $(G' = (V, E'), k')$ mit

$$E' = \{\{v, u\} \mid v, u \in V, v \neq u, \{v, u\} \notin E\}, \quad k' = |V| - k.$$

Damit ist G' der Komplementgraph von G .

a) Geben Sie die Anzahl der Knoten $|V'|$ und die Anzahl der Kanten $|E'|$ der resultierenden VERTEXCOVER-Instanz in Abhängigkeit von der originalen CLIQUE-Instanz an.

b) Beweisen Sie Laufzeit-Lower-Bounds für VERTEXCOVER bezüglich $|V'|$ und $|E'|$ basierend auf der ETH und der oben beschriebenen Reduktion.

Hinweis: Unter Annahme der ETH kann CLIQUE weder in $2^{o(|V|)} \cdot |I|^{O(1)}$ noch in $2^{o(\sqrt{|E|})} \cdot |I|^{O(1)}$ gelöst werden.

Problem: HITTINGSET**Eingabe:** Eine Grundmenge U , Teilmengen $F_1, \dots, F_r \subseteq U$ und eine Zahl k .**Entscheide:** Gibt es $H \subseteq U$ mit $|H| \leq k$ und $H \cap F_i \neq \emptyset$ für alle $i \in [r]$?Betrachten Sie folgende Reduktion von 3-SAT auf HITTINGSET. Für eine 3-SAT-Instanz mit Variablen $x_1, \dots, x_n$ und Klauseln $C_1, \dots, C_m$ setze $U = \{x_i, \bar{x}_i \mid i \in [n]\}$. Für jedes $i \in [n]$ nimm die *Paar-Menge* $\{x_i, \bar{x}_i\} \subseteq U$ hinzu, und für jede Klausel C_j die *Klausel-Menge* $C_j \subseteq U$ (die Menge ihrer Literale). Setze $k = n$.

c) Geben Sie die Größe der Grundmenge $|U|$ und die Anzahl der Teilmengen r der resultierenden HITTINGSET-Instanz in Abhängigkeit von der originalen 3-SAT-Instanz an.

d) Beweisen Sie Laufzeit-Lower-Bounds für HITTINGSET bezüglich $|U|$ und r basierend auf der ETH und der oben beschriebenen Reduktion.

Hinweis: Unter Annahme der ETH ist 3-SAT nicht in $2^{o(n)} \cdot |I|^{O(1)}$ lösbar, mit dem Sparsification-Lemma auch nicht in $2^{o(m)} \cdot |I|^{O(1)}$.

Aufgabe 5 APPROXIMATIVE ALGORITHMEN – MAX-3-SAT (2 + 5 + 3 + (10) Punkte)

Problem: MAX-3-SAT

Eingabe: Eine Formel φ in konjunktiver Normalform mit m Klauseln, jede Klausel besteht aus (bis zu) drei Literalen.

Ausgabe: Eine Belegung der Variablen, welche die Anzahl $v(\cdot)$ der erfüllten Klauseln maximiert.

Betrachten Sie den folgenden Algorithmus:

```
Algorithmus  $A(\varphi)$ 
1  $\beta_0 \leftarrow$  Belegung, die jede Variable auf falsch setzt
2  $\beta_1 \leftarrow$  Belegung, die jede Variable auf wahr setzt
3 Werte  $v(\beta_0)$  und  $v(\beta_1)$  aus (Anzahl der jeweils erfüllten Klauseln)
4 return die Belegung mit größerem Wert (bei Gleichstand  $\beta_0$ )
```

- a) Wenden Sie den Algorithmus A auf folgende Formel mit den Variablen x_1, x_2, x_3 an:

$$\varphi = (x_1 \vee x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_2 \vee \neg x_3).$$

Geben Sie $v(\beta_0)$, $v(\beta_1)$, den Wert der Ausgabe von A sowie den Optimalwert $v(\text{OPT}(\varphi))$ an.

- b) Beweisen Sie, dass der Algorithmus A die Güte 2 hat, also $v(A(\varphi)) \geq \frac{1}{2} v(\text{OPT}(\varphi))$ für jede Eingabe φ gilt.
- c) Geben Sie eine (scharfe) Instanz an, bei welcher der Algorithmus A genau die Hälfte des Optimums erfüllt, also $v(A(\varphi)) = \frac{1}{2} v(\text{OPT}(\varphi))$ gilt. Begründen Sie Ihre Antwort.
- d) **BONUS:** Sei $k \in \mathbb{N}_{\geq 2}$. Wir betrachten MAX- k -SAT, bei dem jede Klausel aus *genau* k Literalen mit paarweise verschiedenen Variablen besteht. Welche Güte garantiert derselbe Algorithmus A für MAX- k -SAT? Beweisen Sie diese Güte und zeigen Sie, dass sie scharf ist.