Technische Universität Wien
Institut für Computergraphik und Algorithmen
Arbeitsbereich für Algorithmen und Komplexität

186.866 Algorithmen und Datenstrukturen VU

Übungsblatt 5

für die Übung am Montag den 28. bzw. Dienstag den 29. Mai 2018.

Geben Sie bis **spätestens Sonntag, 27.5.2018, 13:00 Uhr** über TUWEL an, welche Beispiele Sie bearbeitet und gelöst haben. Gehen Sie dabei folgendermaßen vor:

- TUWEL (`https://tuwel.tuwien.ac.at`)
  Kurs *186.866 Algorithmen und Datenstrukturen (VU 5.5)* im Abschnitt *Übungsblätter*
- Bearbeitete Beispiele ankreuzen **und** abgeben
  - Link *Ankreuzen Übungsblatt 5*
    Button *Abgabe bearbeiten*
    Bearbeitete Beispiele anhaken und *Änderungen speichern*.
  - Link *Hochladen Lösungen Übungsblatt 5*
    Button *Abgabe hinzufügen*
    PDF-Datei mit Lösungen hochladen und *Änderungen sichern*.

Bitte beachten Sie:

- Sie können **vor** der Deadline beliebig oft ihre Auswahl an Beispielen und das zugehörige Lösungs-PDF verändern, aber **nach** der Deadline gibt es **keine** Veränderung Ihrer angekreuzten Beispiele bzw. der abgegebenen Dateien!
- Sie können Ihre Lösungen entweder direkt in einem Textverarbeitungsprogramm erstellen, oder aber auch gut leserliche Scans bzw. Fotos von handschriftlichen Ausarbeitungen einreichen.
- Bitte geben Sie Ihren Namen, Matrikelnummer und E-Mail-Adresse in den Ausarbeitungen an.
- Beachten Sie die Richtlinien für das An- und Aberkennen von Beispielen in den Vorbesprechungsfolien. Neben der Überprüfung in der Übungseinheit werden danach stichprobenartig weitere Abgaben auf Spekulation und Plagiate überprüft.

Aufgabe 1

(a) Sei $\mathcal{P}$ ein Ja/Nein-Problem für Instanzen mit Größe n. Nehmen Sie an, dass es eine Reduktion von $\mathcal{P}$ auf ein Problem $\mathcal{Q}$ gibt, die $O(n^3 \cdot \log n)$ Zeit benötigt. Nehmen Sie weiterhin an, dass Sie Problem $\mathcal{Q}$ in Zeit $O(m^2)$ lösen können, wobei m nun die Eingabegröße einer Instanz von $\mathcal{Q}$ ist.

- Bedeutet dies, dass jede Instanz von $\mathcal{P}$ in polynomieller Zeit gelöst werden kann?
- Geben Sie eine obere Schranke für die Laufzeit (eines optimalen Algorithmus) für das Problem $\mathcal{P}$ an und begründen Sie Ihre Antwort kurz.

(b) Sei $\mathcal{P}^*$ ein Ja/Nein-Problem für Instanzen mit Größe n. Nehmen Sie an, dass es eine Reduktion von $\mathcal{P}^*$ auf ein Problem $\mathcal{Q}^*$ gibt, die $O(n)$ Zeit benötigt. Nehmen Sie weiterhin an, dass Problem $\mathcal{Q}^*$ NP-vollständig ist. Welche der folgenden Aussagen sind dann wahr? Begründen Sie Ihre Antwort.

- $\mathcal{P}^*$ ist in NP.
- $\mathcal{P}^*$ ist NP-vollständig.
- Falls Sat in polynomieller Zeit gelöst werden kann, kann auch $\mathcal{P}^*$ in polynomieller Zeit gelöst werden.
- Falls Sat in linearer Zeit gelöst werden kann, kann auch $\mathcal{P}^*$ in linearer Zeit gelöst werden.

Aufgabe 2 Betrachten Sie die Reduktion von 3-Sat auf Independent Set aus der Vorlesung. Zeichnen Sie den Graphen G_ϕ, der als Resultat der Reduktion für die folgende 3-Sat Formel ϕ entsteht. Wie groß muss das größte Independent Set in G_ϕ mindestens sein, damit ϕ erfüllbar ist?

$$\phi = (x_2 \vee \overline{x_1} \vee x_4) \wedge (x_2 \vee \overline{x_3} \vee x_4) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3) \wedge (x_4 \vee \overline{x_1} \vee x_3) \wedge (x_3 \vee \overline{x_4} \vee x_1)$$

Aufgabe 3 Geben Sie für die folgenden Probleme an, ob Sie in NP sind. Falls ja, erklären Sie, wie ein passendes Zertifikat aussehen kann.

- Gegeben ein gewichteter Graph G, eine Zahl k und Knoten u und v, gibt es einen Pfad der Länge k oder kürzer von u nach v?
- Gegeben eine Stellung auf einem $n \times n$ Dame-Brett, gibt es eine Strategie für Weiß, so dass Weiß sicher gewinnt?

- Gegeben eine KNF-Formel ϕ, ist ϕ eine Tautologie, das heißt, ist ϕ für jede Belegung wahr?
- Gegeben zwei natürliche Zahlen m und n, gibt es eine Zahl kleiner n, die m ganzzahlig teilt?
- Gegeben ein Graph G und eine Zahl k, hat G einen Minimal Spanning Tree T, so dass kein Knoten in T Grad größer als k hat?

Aufgabe 4 Sei $G = (V, E)$ ein Graph. Wir nennen eine Menge von Knoten $GS \subseteq V$ ein Guarding Set, falls jeder Knoten in $V \setminus GS$ zu mindestens einem Knoten in GS adjazent ist. Das Guarding Set Problem ist wie folgt definiert:

Gegeben ein Graph $G = (V, E)$ und eine ganze Zahl k. Gibt es eine Guarding Set GS von G, so dass $|GS| \leq k$?

Geben Sie eine polynomielle Reduktion von Guarding Set auf Set Cover an und begründen Sie die Korrektheit Ihrer Reduktion.

Aufgabe 5 Das Optimierungsproblem Min Guarding Set ist wie folgt definiert:

Gegeben ein Graph $G = (V, E)$. Finden Sie ein minimales Guarding Set auf G. Also ein Guarding Set mit der kleinsten Anzahl an Knoten.

Zeigen Sie, dass Min Guarding Set auf Bäumen in polynomieller Zeit gelöst werden kann. Beschreiben Sie dazu einen Algorithmus, der das Problem löst. Argumentieren Sie, warum Ihr Algorithmus in polynomieller Zeit läuft und warum er korrekt ist.

Aufgabe 6 Wir sagen, eine KNF-Formel ist in 1-positiver Form, falls jede Klausel **maximal** ein positives Literal enthält. Zeigen Sie, dass Sat eingeschränkt auf Formeln in 1-positiver Form in polynomieller Zeit gelöst werden kann. Beschreiben Sie dazu einen Algorithmus, der das Problem löst. Argumentieren Sie, warum Ihr Algorithmus in polynomieller Zeit läuft und warum er korrekt ist.

Hinweise:

- Überlegen Sie, welche drei Arten von Klauseln in einer Formel in 1-positiver Form auftreten können.
- Eine Klausel der Form $(\overline{x_1} \vee \overline{x_2} \vee \cdots \vee \overline{x_{k-1}} \vee x_k)$ ist äquivalent zu $((x_1 \wedge x_2 \wedge \cdots \wedge x_{k-1}) \rightarrow x_k)$.
- Versuchen Sie, möglichst wenige Variablen auf wahr zu setzen.
