Algorithmische Graphentheorie
Wintersemester 2014/2015
Thomas Bläsius, Ignaz Rutter

Übungsblatt 7

Besprechung in der Übung am 12. Februar 2015

Aufgabe 1: Unsortierte Zahlen ⋆⋆

Geben Sie eine Ordnung auf den ganzen Zahlen von 1 bis 100 an, sodass es keine elf Zahlen gibt, die aufsteigend oder absteigend sortiert sind. Gibt es eine solche Ordnung auch noch für die Zahlen von 1 bis 101?

Aufgabe 2: Viele Graphklassen ⋆⋆

Ein Graph kann chordal, co-chordal sowie ein Vergleichbarkeitsgraph oder ein co-Vergleichbarkeitsgraph sein. Zeigen Sie, dass diese Eigenschaften unabhängig voneinander sind, indem Sie für jede der 16 möglichen Kombinationen einen Beispielgraphen angeben.

Aufgabe 3: Maximalität in Split-Graphen ⋆

Geben Sie einen Split-Graph $G = (V, E)$ mit einer Zerlegung $V = S + K$ in eine unabhängige Menge S und eine vollständige Menge K an, sodass S keine maximale unabhängige Menge ist. Geben Sie ein weiteres Beispiel an, bei dem K kein maximale Clique induziert.

Aufgabe 4: Hamiltonkreise in Split-Graphen ⋆⋆

Zeigen Sie, dass das Hamiltonkreisproblem in Split-Graphen NP-vollständig ist.

Hinweis: Benutzen Sie, dass das Hamiltonkreisproblem in bipartiten Graphen NP-schwer ist.

Aufgabe 5: Graphische Gradsequenzen ⋆

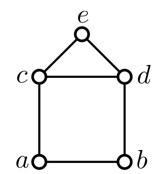
Geben Sie effiziente Algorithmen zur Erkennung von graphischen Gradsequenzen, basierend auf den beiden Charakterisierungen aus der Vorlesung, an. Geben Sie darauf aufbauend einen Algorithmus zur Erkennung von Split-Graphen an.

Aufgabe 6: Gradsequenzen und Bäumen ⋆⋆

Geben Sie einen Algorithmus an, der für eine gegebene Gradsequenz entscheidet, ob es einen Baum mit dieser Gradsequenz gibt. Falls ja, geben Sie einen Baum mit der gegebenen Gradsequenz an.

Aufgabe 7: Permutationsgraphen ⋆

Sei H der nebenstehende Hüttengraph. Zeigen Sie, dass H ein Permutationsgraph ist, indem Sie eine Matching-Repräsentation für H angeben. Zeigen Sie außerdem, dass H und sein Komplement $\bar{H}$ Vergleichbarkeitsgraphen sind.



Hat H auch eine Matching-Repräsentation, bei der die Knoten in der oberen Zeile die Reihenfolge a, b, c, d, e haben?

Aufgabe 8: Permutationsbeschriftung ⋆⋆⋆

Sei $G = (V, E)$ ein Graph und sei $L\colon V \to \{1, 2, \ldots, n\}$ eine bijektive Knotenbeschriftung. Dann ist L eine *Permutationsbeschriftung*, wenn eine Bijektion $L'\colon V \to \{1, 2, \ldots, n\}$ existiert, sodass zwei Knoten x und y genau dann adjazent sind, wenn $\big(L(x) - L(y)\big) \cdot \big(L'(x) - L'(y)\big) < 0$.

Zeigen Sie, dass L genau dann eine Permutationsbeschriftung ist, wenn $F\colon x \mapsto L(x) - d^-(x) + d^+(x)$ injektiv ist, wobei $d^-(x) = |\{y \in \mathrm{Adj}(x) \mid L(y) < L(x)\}|$ und $d^+(x) = |\{y \in \mathrm{Adj}(x) \mid L(y) > L(x)\}|$.

Hinweis: Setzen Sie $L' = F$.

Aufgabe 9: Sortieren mit Stacks ⋆⋆

In der Vorlesung wurde die Sortierung einer Permutation π mittels k paralleler Warteschlangen (Queues) untersucht. Betrachten Sie analog dazu die Sortierung einer Permutation mit k parallelen Stacks, die nach dem LIFO-Prinzip (last-in-first-out) arbeiten.

Zeigen Sie, dass die folgenden Zahlen gleich sind, wenn alle `push`-Operationen zu Beginn und alle `pop`-Operationen am Ende stattfinden müssen.

(a) Die Clique-Cover-Zahl von $G[\pi]$,

(b) die minimale Zahl von Stacks um π zu sortieren,

(c) die Länge der längsten aufsteigenden Teilfolge in π.

Gilt die Gleichheit auch ohne diese Einschränkung?