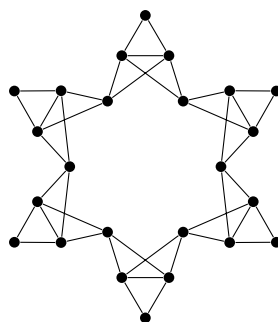


## 9. Übungsblatt zur Vorlesung Algorithmische Graphentheorie (Sommer 2026)

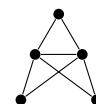
### Aufgabe 1 – Dreifärbbarkeit

Als *Dreifärbung* eines ungerichteten Graphen  $G = (V, E)$  bezeichnen wir eine Funktion  $f: V \rightarrow \{r, g, b\}$ , bei der für jede Kante  $uv \in E$  die Bedingung  $f(u) \neq f(v)$  erfüllt ist, d. h. zwei benachbarte Knoten erhalten niemals dieselbe Farbe. Das Dreifärbbarkeitsproblem 3COL besteht darin zu entscheiden, ob ein gegebener Graph *dreifärbbar* ist, d. h. ob es eine Dreifärbung des Graphen gibt. Dieses Problem ist NP-vollständig.

- a) Finden Sie eine Dreifärbung des Sterngraphen aus Abbildung 1a. **1 Punkt**



(A) Sterngraph



(B) Turmgraph

ABBILDUNG 1: Graphen für Aufgabe 1.

- b) Der Sterngraph aus Teilaufgabe a) fñgt sich aus sechs Exemplaren des Turmgraphen aus Abbildung 1b zusammen. Zeigen Sie, dass auch Sterne, die aus mehr oder weniger als sechs Türmen bestehen, dreifärbbar sind und dass bei einer Dreifärbung alle Knoten mit Grad 2 dieselbe Farbe haben. **2 Punkte**
- c) Das Problem 3COL4 besteht darin zu entscheiden, ob ein gegebener Graph mit Maximalgrad 4 dreifärbbar ist. Zeigen Sie, dass schon 3COL4 NP-schwer ist, indem Sie eine Polynomialzeit-Reduktion von 3COL auf 3COL4 angeben.

Geben Sie also ein Verfahren an, mit der man für einen beliebigen Graphen entscheiden kann, ob er dreifärbbar ist, indem Sie einen (hypothetischen) Test `Test3COL4` auf Dreifärbbarkeit von Graphen mit Maximalgrad 4 nutzen. Zeigen Sie, dass Ihr Verfahren korrekt ist. **4 Punkte**

*Hinweis:* Verwenden Sie Teilaufgabe b)!

## Aufgabe 2 – Chromatische Zahl

Sei  $G = (V, E)$  ein ungerichteter Graph. Die kleinste Zahl  $k$ , die eine  $k$ -Färbung von  $G$  erlaubt, heißt *chromatische Zahl*  $\chi(G)$ .

- a) Sei  $\Delta(G) := \max_{v \in V} \deg v$  der maximale Knotengrad in  $G$ . Zeigen Sie, dass für die chromatische Zahl von  $G$  gilt, dass  $\chi(G) \leq \Delta(G) + 1$ . **2 Punkte**
- b) Zeigen Sie, dass die Ungleichung aus Teilaufgabe a) scharf ist. Geben Sie dazu für jedes  $n \in \mathbb{N}$  und jedes  $\Delta < n$  einen Graphen  $G_{n,\Delta}$  an, der  $n$  Knoten besitzt, Maximalgrad  $\Delta$  aufweist und  $\chi(G_{n,\Delta}) = \Delta + 1$  erfüllt. **2 Punkte**
- c) Erstellen Sie ein kommentiertes ganzzahliges lineares Programm (ILP), welches  $\chi(G)$  für einen gegebenen Graphen  $G = (V, E)$  ermittelt. Sie dürfen Nebenbedingungen verwenden, die das Ungleich-Symbol ( $\neq$ ) enthalten. Verwenden Sie Ganzzahlen für die Farben. **2 Punkte**
- d) Eigentlich dürfen lineare Programme keine Nebenbedingungen mit dem Ungleich-Symbol ( $\neq$ ) enthalten. Im Fall von **ganzzahligen** linearen Programmen erlauben wir sie aber, da wir mithilfe von Hilfsvariablen eine äquivalente Formulierung erstellen können, die nur kleinergleich ( $\leq$ ) und größergleich ( $\geq$ ) enthält.

Finden Sie für das folgende (simple) ganzzahlige lineare Programm, wobei  $A$  und  $B$  Konstanten sind, eine äquivalente Formulierung, welche nur  $\leq$  und  $\geq$ , nicht aber  $\neq$  enthält. Begründen Sie die Korrektheit ihres Ansatzes.

<b>Variablen:</b>	$x_1, x_2 \in \mathbb{N}_0^+$
<b>Zielfunktion:</b>	Minimiere $Ax_1 + Bx_2$
<b>Nebenbedingung:</b>	$x_1 \neq x_2$

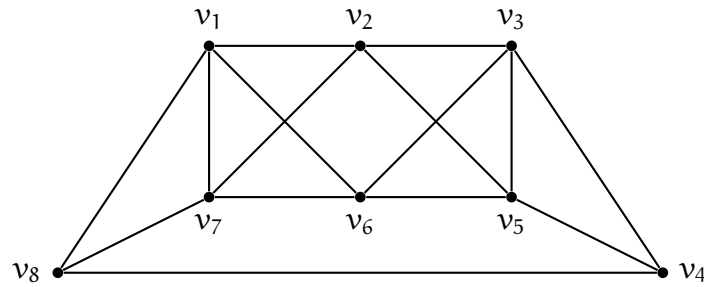
**3 Punkte**

*Tipp:* Führen Sie eine binäre Hilfsvariable ein, die angibt, ob  $x_1 < x_2$  oder  $x_1 > x_2$ . Nehmen Sie zudem an, dass es eine große Konstante  $M$  gibt, für die gilt  $M > x_1, x_2$ . Die konkreten Konstanten  $A$  und  $B$  sind für die Umformung nicht relevant.

PuLP

- e) Implementieren Sie das ILP aus Teilaufgabe (c) mit PuLP. Ihr Modell muss in der Lage sein, ungerichtete Graphen beliebiger Größe einzulesen. Wählen Sie dazu eine geeignete Darstellung des Graphen. Laden Sie Ihre Lösung auf WueCampus hoch.

Verwenden Sie in Ihrer Abgabe den folgenden Beispielgraphen. Was ist die chromatische Zahl dieses Graphen?



**3 Zusatzpunkte**

- f) Erstellen Sie nun ein kommentiertes ILP, welches  $\chi(G)$  für einen gegebenen Graphen  $G = (V, E)$  ermittelt, und in dem alle bis auf höchstens eine der ganzzahligen Variablen  $\in \{0, 1\}$  sind.

**4 Zusatzpunkte**

*Hinweis:* Nutzen Sie als die eine ganzzahlige Variable, die nicht in  $\{0, 1\}$  liegen muss, die Gesamtzahl an Farben.

PuLP

- g) Implementieren Sie das ILP aus Teilaufgabe (f) in PuLP. Ihr Modell muss wieder in der Lage sein, beliebige ungerichtete Graphen einzulesen. Testen Sie Ihre Implementierung anhand des Graphen aus Teilaufgabe (e). Laden Sie Ihren Code in WueCampus hoch.

**3 Zusatzpunkte**

PuLP

- h) Vergleichen Sie die Laufzeiten der Implementierungen aus den Teilaufgaben (e) und (g) auf einer aussagekräftigen Anzahl an (zufällig generierten) Graphen miteinander. Welche Graphen haben Sie verwendet/erzeugt? Wie verhält sich die Laufzeit in Abhängigkeit von der Knotenanzahl? Laden Sie eine Übersicht der Laufzeiten und Ihre Dateien zum Erzeugen/Importieren der Graphen auf WueCampus hoch.

**4 Zusatzpunkte**

### Aufgabe 3 – Perfektes Eliminationsschema

In der Vorlesung wurde bewiesen, dass die Graphen mit perfektem Eliminationsschema gerade die chordalen Graphen sind. Man kann ein perfektes Eliminationsschema eines chordalen Graphen mit  $n$  Knoten in  $O(n^4)$  Zeit bestimmen, indem man den Beweis zum Satz von Dirac, um simplizialen Knoten zu finden, implementiert (das geht auf direkte Weise in  $O(n^3)$  Zeit) und dann den gefunden Knoten löscht. Man wiederholt den Algorithmus  $n$  Mal bis alle Knoten entfernt wurden.

Zeigen Sie, wie man ein perfektes Eliminationsschema eines chordalen Graphen mit  $n$  Knoten in  $O(n^4)$  Zeit bestimmen kann. **4 Punkte**

*Tipp:* Setzen Sie den Beweis von Dirac (siehe VL-Folien!) durch einen rekursiven Algorithmus um. Der Beweis geht von einer Menge  $U \subsetneq V$  maximaler Kardinalität aus, für die gelten muss, dass  $G[U]$  zusammenhängend ist und  $U \cup N(U) \neq V$  gilt. Argumentieren Sie, dass man die Maximalität von  $U$  durch ein schwächeres Kriterium ersetzen kann, das sich leichter überprüfen lässt.

---

Bitte geben Sie Ihre Lösungen bis **Dienstag, 23. Juni 2026, 13:00 Uhr** einmal pro Gruppe über WueCampus als PDF-Datei ab. Geben Sie stets die Namen aller an, die das Übungsblatt bearbeitet haben (max. 2).

Begründen Sie Ihre Behauptungen und kommentieren Sie Ihren Pseudocode!

Aufgaben, die mit PuLP gekennzeichnet sind, fordern das Erstellen und Lösen von linearen Programmen. Geben Sie Ihren kommentierten zusammen mit Ihrer Bearbeitung des Übungsblatts auf WueCampus ab. Der Quellcode muss von derselben Person abgegeben werden, die auch das PDF hochgeladen hat.

Plagiate werden mit 0 Punkten für das ganze Übungsblatt gewertet.