Abteilung Algorithmik **Winter 2020/21**
Institut für Betriebssysteme und Rechnerverbund
TU Braunschweig

Dr. Linda Kleist
Phillip Keldenich
Dominik Krupke

Mathematische Methoden der Algorithmik
Übungsblatt 3 vom 18. November 2020

Die Abgabe eurer Lösungen zu diesem Blatt muss bis zum 2. Dezember 2020 um 16:00 Uhr per E-Mail an `sven.langner@tu-bs.de` erfolgen. Am besten gebt ihr eure Lösung als einzelnes PDF-Dokument (LaTeX, Word oder Scan) ab. Ihr könnt auch handschriftliche Lösungen scannen oder abfotografieren; nach Möglichkeit fasst ihr auch in diesem Fall am besten die einzelnen Seiten zu einem PDF-Dokument zusammen. Achtet dabei bitte einerseits auf Lesbarkeit und andererseits auf eine vernünftige Dateigröße (höchstens etwa 2 MB/Seite). Vermeidet bitte auch das Versenden von ZIP-Archiven, da es hier möglicherweise zu Problemen mit dem Virenscanner auf dem Mailserver der TU kommen kann.

Aufgabe 0 (Prüfungsfragen): Die kleine Übung soll auch als kontinuierliche Vorbereitung auf die Prüfung dienen, indem jeweils mögliche Prüfungsfragen zu den aktuellen Themen diskutiert werden.

Überlegt euch in eurer Hausaufgabengruppe 2 leichte, 2 mittlere und 2 schwere mögliche Prüfungsfragen zum Stoff der Vorlesungen 3–5 und gebt diese mit eurer Abgabe ab. Die Fragen werden bis zur kleinen Übung zusammengetragen und die besten Fragen werden dann kollaborativ in der kleinen Übung diskutiert.

Aufgabe 1 (Formen von LPs): Betrachte das folgende lineare Optimierungsproblem:

$$(P)\left\{\begin{array}{lrcrcr}
\min & x_1 & + & x_2 & & \\
\text{s. t.} & x_1 & - & 3x_2 & \geq & -9 \\
 & 2x_1 & + & x_2 & \geq & 8 \\
 & 2x_1 & + & 5x_2 & \leq & 20 \\
 & x_1 & & & \leq & 10 \\
 & x_1 & & & \geq & 0
\end{array}\right.$$

Transformiere das lineare Programm P in die folgenden Formen.

(a) $\min\{c^T x \mid Ax \leq b, x \geq 0\}$

(b) $\max\{c^T x \mid Ax = b, x \geq 0\}$

(c) $\min\{c^T x \mid Ax \leq b, x \text{ frei}\}$.

Die Matrixschreibweise ist nicht gefordert. **(10 Punkte)**

Aufgabe 2 (Basislösungen): Betrachte das folgende lineare Programm

$$(P) = \min\{c^T x \mid Ax = b, x \geq 0\} \text{ mit: } A = \begin{pmatrix} 1 & 0 & 2 & 1 & 4 \\ 0 & 4 & 0 & 4 & 0 \\ 0 & 0 & 1 & 0 & 2 \end{pmatrix}, b = \begin{pmatrix} 5 \\ 7 \\ 1 \end{pmatrix}, c = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Bestimme die Basislösungen von (P) und ihren Zielfunktionswert. Welche Basislösungen sind zulässig? Welche Basislösung ist die beste? Ist dies eine optimale Lösung von (P), und wenn ja, warum?

Falls eine Teilmenge der Spalten keine Basis ist, darfst du dies ohne Begründung angeben. Du darfst für das Lösen von linearen Gleichungssystemen und das Bestimmen von Inversen auf Tools wie `numpy.linalg.solve` und `numpy.linalg.inv`, WolframAlpha o.Ä. zurückgreifen. **(20 Punkte)**

Aufgabe 3 (Integer Programming – Minimale Spannbäume): Beim Problem *Minimum Spanning Tree* (MST) suchen wir in einem Graphen $G = (V, E)$ mit Kantengewichten c_e nach einem Baum minimalen Gewichts, der alle Knoten aus V verbindet. Dieses Problem lässt sich wie folgt als ganzzahliges lineares Programm darstellen.

$$\min \sum_{e \in E} c_e \cdot x_e \tag{1}$$

$$\sum_{e \in E} x_e = |V| - 1 \tag{2}$$

$$\sum_{e \in E(S, V \setminus S)} x_e \geq 1 \qquad \forall S \subsetneq V, S \neq \emptyset \tag{3}$$

$$x_e \in \mathbb{B} \qquad \forall e \in E \tag{4}$$

Hier bedeutet $x_e = 1$, dass die Kante e im Baum enthalten ist. Wir minimieren die Summe des Gewichts c_e der verwendeten Kanten. Die beiden Bedingungen geben an, dass eine Lösung mit $|V| - 1$ Kanten suchen, die zusätzlich zusammenhängend ist (was einen Baum beschreibt).

Euch ist vielleicht aufgefallen, dass es exponentiell viele Ungleichungen in diesem Integer Program gibt. Tatsächlich ist dies gewöhnlich kein Problem, solange effizient bestimmt werden kann ob eine beliebige Lösung (fraktional) gültig ist oder ansonsten eine verletzte Bedingungen genannt werden kann. In der Praxis würde man die zusätzlichen Bedingungen dann nach und nach hinzufügen, bis wir eine gültige Lösung erhalten.

Ändere die Formulierung nun so ab, dass stattdessen das Gewicht der schwersten Kante im Baum minimiert wird. **(10 Punkte)**

Aufgabe 4 (Mixed Integer Programming – Höhenbegrenzte Spannbäume): Beim Problem *Minimum Bounded Height Spanning Tree* (MBHST) besteht der Input aus einem Graphen $G = (V, E)$ mit Kantengewichten $c_e \geq 0$ für alle $e \in E$ und einer gegebenen Wurzel $v_r \in V$, sowie einer Höhenschranke $\ell \geq 0$. Das Ziel ist es, einen Spannbaum T mit minimalem Gesamtgewicht zu bestimmen, bei dem die Höhe durch ℓ beschränkt ist, d.h., dass für jeden Knoten $v \in V$ die Summe der Gewichte der Kanten auf dem eindeutigen Pfad von v_r nach v höchstens ℓ ist.

Hinweise:

1 Nutze eine zusätzliche fraktionale Variable für jeden Knoten.

2 Nutze gerichtete Kanten.

3 Informiere dich zur sogenannten Big-M-Methode.

(20 Punkte)