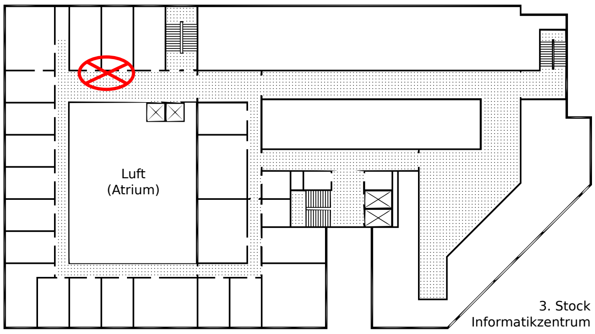
Abteilung Algorithmik **Winter 2019/20**
Institut für Betriebssysteme und Rechnerverbund
TU Braunschweig

Dr. Linda Kleist
Phillip Keldenich
Dominik Krupke

Mathematische Methoden der Algorithmik
Übungsblatt 6 vom 22.01.2020

Die Abgabe eurer Lösungen zu diesem Blatt sind bis Mittwoch, den 05.02.2020 um 14:50 im Hausaufgabenschrank der Algorithmik möglich (siehe Skizze) oder direkt in der kleinen Übung.

Bitte die Blätter vorne deutlich mit eigenem Namen sowie Matrikelnummer versehen und zusammenheften!



Aufgabe 1 (Column Generation): Für viele Optimierungsprobleme brauchen wir am Ende nur einen Bruchteil der Variablen. Oft lässt sich anhand der Kostenfunktion auch gut vorhersagen, welche Variablen für die optimale Lösung nützlich sein könnten.

Betrachte das Minimum Weight Perfect Matching Problem. Für einen Graphen $G = (V, E)$ soll ein perfektes Matching gefunden werden, sodass die Summe der Gewichte $d(v, w)$ aller gematchten Knotenpaare $v, w \in V$ minimal ist. Das Linear Program zu dem Problem ist

$$\begin{aligned} \min \quad & \sum_{vw \in E} x_{vw} \cdot d(v, w) \\ \text{s.t.} \quad & \sum_{vw \in E(v)} x_{vw} = 1 \quad \forall v \in V \\ & x_{vw} \geq 0 \quad \forall vw \in E \end{aligned}$$

und das zugehörige duale Problem

$$\begin{aligned} \max \quad & \sum_{v \in V} y_v \\ \text{s.t.} \quad & y_v + y_w \leq d(v, w) \quad \forall vw \in E \end{aligned}$$

a) Gegeben sei der Graph $G = (\{v_0, v_1, v_2, v_3, v_4, v_5\}, \{v_0v_1, v_0v_5, v_2v_3, v_4v_5\})$ mit

$$d(v_0, v_1) = 1.1, d(v_2, v_3) = 1.0, d(v_4, v_5) = 1.2, \text{ und } d(v_0, v_5) = 1.0.$$

Gebe eine optimale primale und duale Lösung an. Hinweis: Dies benötigt keinen Simplex.

b) Das eigentliche Problem hat noch einige weitere Kanten, aber wir wissen, dass diese alle mindestens das Gewicht 2.41 haben. Was passiert im primalen und dualen Problem wenn wir eine weitere Kante zum Graphen hinzufügen? Argumentiere über die duale Lösung, warum wir diese zusätzlichen Kanten nicht betrachten müssen (also keine dieser Kanten unsere optimale Lösung verändert).

c) Was passiert im primalen und dualen Problem wenn wir die Kante v_1v_4 mit $d(v_1, v_4) = 1.0$ hinzufügen? Gebe die zugehörige optimale primale und duale Lösung an.

d) Wie kompliziert ist es eine neue Variable zu einem bereits gelöstem Simplex Tableau hinzuzufügen und dieses zu optimieren? (A_B^{-1} des ehemals optimalen Tableaus wurde gespeichert.)

(20 Punkte)

Aufgabe 2 (Gomory Cuts): Löse das folgende IP mittels Simplex, indem du iterativ Gomory Cuts auf Zeilen mit fraktionaler rechter Seite hinzufügst und dann mittels dualen Simplexschritten auf dem Tableau weiter arbeitest.

$$\max x_1 + x_2$$

$$-x_1 + 3x_2 \leq 6$$

$$3x_1 + x_2 \leq 9$$

$$x_1, x_2 \geq 0, \text{ integral}$$

Zeichne das zweidimensionale Polytop über x_1, x_2. Zeichne alle hinzugefügten Gomory Cuts ein und kennzeichne sie.

Ein Gomory Cut erhälst du wie folgt:

- Betrachte eine Zeile $\sum_i a_{ji} \cdot x_i = b_j$ mit fraktionaler rechter Seite b_j.
- Offensichtlich ist $\sum_i \lfloor a_{ji} \rfloor \cdot x_i \leq \lfloor b_j \rfloor$ eine ebenfalls gültige Bedingung, wenn $x \geq 0$ integral ist.
- Wenn wir dies von der ursprüngliche Zeile abziehen, erhalten wir $\sum_i (a_{ji} - \lfloor a_{ji} \rfloor) \cdot x_i \geq b_j - \lfloor b_j \rfloor$ die wir mit einer weiteren negativen Slackvariablen zu einer Gleichheitsbedingung umformen können.
- Im Tableau fügen wir also eine Zeile und Spalte hinzu.
- Wenn wir diese Zeile nun negieren, erhalten wir mit der Slackvariable automatisch eine Basis (mit negativer rechter Seite) und können mit einem dualen Simplex-Schritt fortfahren.

(20 Punkte)

Aufgabe 3 (Praktisches Problem): Du willst dir einen neuen Computer bauen. Er soll maximal 700 Euro kosten und die maximale Leistung liefern (Summe der Leistung aller Komponenten). Desweiteren kannst du dein altes Netzteil nur weiterverwenden, wenn der Computer weniger als $350W$ verbraucht. Ansonsten musst du für 100 Euro ein stärkeres Netzteil kaufen.
Schreibe das Problem als IP auf und löse es mit `http://hgourvest.github.io/glpk.js/`.

- Mainboard:

Konfiguration:	Preis:	Verbrauch:	Leistung:
MB_A	70	20	70
MB_B	50	80	100
MB_C	100	40	100

- CPU:

Konfiguration:	Preis:	Verbrauch:	Leistung:
C_1	100	50	100
C_2	200	75	350
C_3	100	55	90
C_4	150	80	175
C_5	200	60	250

- Kompatibilität:

CPU:	MB_A	MB_B	MB_C
C_1	x		
C_2	x		
C_3		x	x
C_4		x	x
C_5		x	x

- Arbeitsspeicher:

Konfiguration:	Preis:	Verbrauch:	Leistung:
16GB	100	20	75
2x16GB	200	40	150
4x8GB	125	55	150

  - 4x8GB wird nur von Mainboard MB_C unterstützt
  - Auf Mainboard MB_A mit CPU C_2 ist die Leistung für jeden Arbeitsspeicher um 20% höher.

- Grafikkarte:

Konfiguration:	Preis:	Verbrauch:	Leistung:
G_1	150	150	100
G_2	250	150	250
G_3	400	250	500

(20 Punkte)