Mathematische Optimierung I
Wintersemester 2006/2007
Abgabe: bis Donnerstag, 16. November 2006, **vor** der Vorlesung

Übungsblatt 3

Aufgabe 9:
Löse das folgende LP mit dem Simplexalgorithmus in Tableauform mit der Indexregel von Bland. Nummeriere die Tableauzeilen durch $(Z_1, Z_2, \ldots)$ und gib für jede neue Zeile an, durch welche Zeilentransformation sie entsteht (in der Form „$Z_3 - \frac{4}{3}Z_1$“). Mache deutlich, welchen Variablen die Spalten entsprechen. Gib zu jedem Tableau die Basis an und notiere, welche Variable die Basis verlässt und welche hinzukommt.

$$\max \Big\{ 5x_1 + 4x_2 : \begin{array}{rcrcr} x_1 & + & 2x_2 & \leq & 6 \\ -2x_1 & + & x_2 & \leq & 4 \\ 5x_1 & + & 3x_2 & \leq & 15 \\ & & x_1,\ x_2 & \geq & 0 \end{array} \Big\}$$

Vergiss nicht, die Lösungen des LPs in den Variablen der Aufgabenstellung sowie die optimalen Zielfunktionswerte anzugeben! (4 Punkte)

Aufgabe 10:
Gib ein (möglichst einfaches) Beispiel dafür an, dass das duale LP eines unzulässigen LPs ebenfalls unzulässig sein kann. (4 Punkte)

Aufgabe 11:
Sei P ein Polyeder. Ein *Extrempunkt* von P ist ein Punkt $x \in P$, der sich nicht als Konvexkombination von x verschiedener Punkte aus P darstellen lässt, für den es also keine $x_1, \ldots, x_k \in P \setminus \{x\}$ und $\lambda_1, \ldots, \lambda_k \geq 0$ mit $\lambda_1 + \ldots + \lambda_k = 1$ gibt ($k \in \mathbb{N}$), so dass $x = \lambda_1 x_1 + \ldots + \lambda_k x_k$.
Zeige, dass die Menge der Extrempunkte von P und die Menge der Ecken von P gleich sind. (4 Punkte)

Aufgabe 12:
Im Job-Assignment-Problem sind n Jobs mit Zeitbedarf $t_1, \dots, t_n \in \mathbb{R}_+$ und m Arbeiter gegeben. Die Menge $S_i \subseteq \{1, \dots, m\}$ gibt an, welche Arbeiter den Job i ausführen können ($i \in \{1, \dots, n\}$). An jedem der Jobs können gleichzeitig beliebig viele Arbeiter aus S_i arbeiten. Ein Arbeiter kann an verschiedenen Jobs arbeiten, jedoch nur nacheinander. Das folgende LP minimiert die Gesamtzeit für die Erledigung aller Jobs; dabei bezeichnet die Variable x_{ij} jeweils die Zeit, die Arbeiter j insgesamt an Job i arbeitet.

$$\min \Big\{ T : \begin{array}{ll} x_{ij} \geq 0 & (i \in \{1, \dots, n\}, j \in S_i) \\ \sum_{j \in S_i} x_{ij} = t_i & (i \in \{1, \dots, n\}) \\ \sum_{i : j \in S_i} x_{ij} \leq T & (j \in \{1, \dots, m\}) \Big\} \end{array}$$

a) Stelle das duale LP auf.

b) Gib für $n = 2$ und $t_1, t_2 > 0$ einen direkten Algorithmus an, der eine optimale primale und eine optimale duale Lösung findet, und begründe seine Korrektheit.

(4 Punkte)

Aufgabe 13:
Implementiere den Simplexalgorithmus.
Der Algorithmus bekommt keine Startlösung, sondern soll selbst eine Startbasis finden. Die Aufgabe besteht darin, entweder Unzulässigkeit oder Unbeschränktheit festzustellen und einen Vektor auszugeben, der dies beweist, oder eine optimale Lösung mitsamt Zielfunktionswert auszugeben, nebst dualer Lösung. So kann die Korrektheit des Outputs in jedem Fall leicht geprüft werden. Im Programmtext sollen geeignete Kommentare stehen.
Als Input erwartet das Programm eine Textdatei, die eine Linear-Programming-Instanz $\max\{cx : x \in P\}$ mit $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ beschreibt, wobei A eine $m \times n$-Matrix ist. In der ersten Zeile stehen dabei m und n, in der zweiten Zeile die n Einträge von c und in der dritten Zeile die m Einträge von b^T. Die nächsten m Zeilen entsprechen den Zeilen von A. Das Programm sollte mit „`Programmname Instanzdateiname`“ aufgerufen werden können. Auf der Webseite für die Übungen werden Beispielinstanzen und eine Einleseroutine (in der Sprache C) zur Verfügung gestellt.
Probleme numerischer Stabilität können hier ignoriert werden, aber zumindest kleine LPs (mit $n, m < 10$) mit ganzzahligen, einstelligen Inputzahlen sollten annähernd korrekt gelöst werden.
Es soll mit dem GNU-C/C++-Compiler bzw. mit dem aktuellen Java Development Kit kompilierbarer Quellcode abgegeben werden. Bitte schicke Deinen Code bis spätestens 16.11.06, 12:00 Uhr, per E-Mail an `mueller@or.uni-bonn.de`.
(16 Punkte)