Effiziente Algorithmen – Gruppe A

Aufgabe 1A (10 Punkte)

Betrachten Sie das folgende Online-Problem Bin-Packing:
Zur Verfügung stehen unendlich viele Container der Kapazität $C \in \mathbf{N}$. Ihr Algorithmus bekommt in jedem Schritt i ein Objekt p_i der Größe $c_i \in \mathbf{N}$ und muß dieses Objekt sofort in einem der Container ablegen. Dabei darf die Belegung eines Containers seine Kapazität nicht überschreiten, wobei die Belegung eines Containers die Summe der Größen der Objekte in diesem Container ist. Objekte können nicht aufgeteilt werden. Einmal abgelegte Objekte dürfen nicht mehr bewegt werden. Das nächste Objekt erhält der Algorithmus erst, nachdem er das vorherige Objekt in einem der Container abgelegt hat.
Die Kosten einer Lösung sind die Anzahl der benutzten (nicht leeren) Container, die es zu minimieren gilt. Entwerfen Sie einen 2-kompetetiven Algorithmus und beweisen Sie diese Güte.

Aufgabe 2A (10 Punkte)

Betrachten Sie das folgende Optimierungsproblem 3-Hitting Set:

Eingabe:	Eine Grundmenge $U = \{1, \dots, n\}$, eine Familie $\mathcal{S} = \{S_1, \dots, S_m\}$ von Teilmengen $S_i \subseteq U$ mit $\lvert S_i \rvert \leq 3$ für alle $1 \leq i \leq m$.
Zulässige Lösungen:	Eine Teilmenge $U' \subseteq U$, so daß U' jedes S_i abdeckt, d.h. $S_i \cap U' \neq \emptyset$ für jedes $1 \leq i \leq m$.
Frage:	Finden Sie zulässige Lösung U' mit minimaler Größe $\lvert U' \rvert$.

Finden Sie einen Algorithmus, der dieses Problem mit konstanter Güte approximiert. Beweisen Sie seine Korrektheit, Laufzeit und Güte.

Aufgabe 3A (10 Punkte)

Es sei das folgende Optimierungsproblem gegeben.
Maximiere $-x + 4y$ unter den Nebenbedingungen:

$$\begin{aligned} x + 2y &\leq 3 & \qquad -3x + y &\leq 10 \\ 2x - y &\geq 3 & x, y &\geq 0 \end{aligned}$$

Finden Sie eine optimale reelle und eine optimale ganzzahlige Lösung mittels eines Verfahrens Ihrer Wahl. Geben Sie Ihren Lösungsweg detailliert an.

Aufgabe 4A (10 Punkte)

Entwerfen Sie einen effizienten Algorithmus für das folgende Problem. Gegeben sind ein ungerichteter Graph $G = (V, E)$, eine Kapazitätsfunktion $c\colon V \to \mathbf{N}$ und eine Knotenteilmenge $C \subseteq V$. Jeder Knoten $v \in C$ kann bis zu $c(v)$ der angrenzenden Kanten *abdecken*. Die Frage ist, ob auf diese Weise *alle* Kanten des Graphen abgedeckt werden können.
Geben Sie einen Beweis für die Korrektheit Ihres Algorithmus an, und zeigen Sie, daß seine Laufzeit polynomiell ist.

RWTH Aachen **SS 2010**
Lehrgebiet Theoretische Informatik Klausur
Kneis—Langer—Rossmanith 26.7.2010

Effiziente Algorithmen – Gruppe B

Aufgabe 1B (10 Punkte)

Betrachten Sie das folgende Optimierungsproblem 3-Hitting Set:

Eingabe:	Eine Grundmenge $U = \{1, \dots, n\}$, eine Familie $\mathcal{S} = \{S_1, \dots, S_m\}$ von Teilmengen $S_i \subseteq U$ mit $\lvert S_i \rvert \leq 3$ für alle $1 \leq i \leq m$.
Zulässige Lösungen:	Eine Teilmenge $U' \subseteq U$, so daß U' jedes S_i abdeckt, d.h. $S_i \cap U' \neq \emptyset$ für jedes $1 \leq i \leq m$.
Frage:	Finden Sie zulässige Lösung U' mit minimaler Größe $\lvert U' \rvert$.

Finden Sie einen Algorithmus, der dieses Problem mit konstanter Güte approximiert. Beweisen Sie seine Korrektheit, Laufzeit und Güte.

Aufgabe 2B (10 Punkte)

Entwerfen Sie einen effizienten Algorithmus für das folgende Problem. Gegeben sind ein ungerichteter Graph $G = (V, E)$, eine Kapazitätsfunktion $c \colon V \to \mathbf{N}$ und eine Knotenteilmenge $C \subseteq V$. Jeder Knoten $v \in C$ kann bis zu $c(v)$ der angrenzenden Kanten *abdecken*. Die Frage ist, ob auf diese Weise *alle* Kanten des Graphen abgedeckt werden können.

Geben Sie einen Beweis für die Korrektheit Ihres Algorithmus an, und zeigen Sie, daß seine Laufzeit polynomiell ist.

Aufgabe 3B (10 Punkte)

Es sei das folgende Optimierungsproblem gegeben.

Maximiere $4x - y$ unter den Nebenbedingungen:

$$\begin{aligned} 2x + y &\leq 3 & \qquad x - 3y &\leq 10 \\ -x + 2y &\geq 3 & x, y &\geq 0 \end{aligned}$$

Finden Sie eine optimale reelle und eine optimale ganzzahlige Lösung mittels eines Verfahrens Ihrer Wahl. Geben Sie Ihren Lösungsweg detailliert an.

Aufgabe 4B (10 Punkte)

Betrachten Sie das folgende Online-Problem Min-Processors:

Zur Verfügung stehen unendlich viele Prozessoren, aber eine obere Laufzeitschranke $L \in \mathbf{N}$. Ihr Algorithmus bekommt in jedem Schritt i einen Job j_i der Laufzeit $l_i \in \mathbf{N}$ und muß diesen Job sofort einem Prozessor zuweisen.

Dabei darf die Laufzeit aller Jobs auf einem Prozessor die obere Laufzeitschranke L nicht überschreiten. Jeder Job muß vollständig laufen. Einmal zugewiesene Jobs dürfen nicht mehr einem anderen Prozessor zugewiesen werden. Den nächsten Job erhält der Algorithmus erst, nachdem er den vorherigen Job einem Prozessor zugewiesen hat.

Die Kosten einer Lösung sind die Anzahl der verwendeten Prozessoren, die es zu minimieren gilt. Entwerfen Sie einen 2-kompetetiven Algorithmus und beweisen Sie diese Güte.