Rainer Feldmann
Joachim Gehweiler, Nicolai Hähnle,
Frank Hellweg, Martina Hüllmann,
Ulf Lorenz, Daniel Warner

Paderborn, 26.01.2007
Abgabe keine

Präsenzübung zu

# Einführung in Berechenbarkeit, Komplexität und formale Sprachen

WS 2006/2007

Blatt 12

**Aufgabe 29:** Beim Problem *Maschinen-Scheduling* sind zwei identische Maschinen, $n$ Jobs mit Bearbeitungszeiten $t_1, \ldots, t_n$ sowie ein Zeitpunkt $d$ gegeben. Jeder der Jobs kann auf beiden Maschinen ausgeführt werden. Zur Ausführung des $i$-ten Jobs wird dabei auf beiden Maschinen Zeit $t_i$ benötigt. Jede der beiden Maschinen kann zu jedem Zeitpunkt nur einen Job ausführen. Zu entscheiden ist, ob die Jobs so auf die beiden Maschinen verteilt werden können, dass alle Jobs zum Zeitpunkt $d$ abgearbeitet wurden. D.h., zu entscheiden ist, ob zwei Mengen $J_1, J_2$ existieren mit $J_1 \cup J_2 = \{1, \ldots, n\}$ und

$$\sum_{i \in J_1} t_i \leq d \quad \text{und} \quad \sum_{i \in J_2} t_i \leq d.$$

(i) Zeigen Sie, dass *Maschinen-Scheduling* in NP liegt.

(ii) Zeigen Sie, dass sich das folgende Problem *Partition* polynomiell auf *Maschinen-Scheduling* reduzieren lässt.

Bei *Partition* sind $n$ natürliche Zahlen $a_1, \ldots, a_n$ gegeben. Zu entscheiden ist, ob es eine Teilmenge $P \subset \{1, \ldots, n\}$ gibt, so dass

$$\sum_{i \in P} a_i = \sum_{i \notin P} a_i.$$

**Aufgabe 30:** Beim Problem *Benteler* sind $m$ Stahlrohre mit den Längen $l_1, \ldots, l_m$ auf Lager und es liegen $n$ Bestellungen mit den Wunschlängen $b_1, \ldots, b_n$ vor. Zu entscheiden ist, ob die $m$ Stahlrohre so zersägt werden können, dass alle $n$ Kundenwünsche erfüllt sind und kein Verschnitt übrig bleibt. Weiter betrachten wir die Sprache

$$\text{SubsetSum} := \left\{ \langle S, t \rangle \;\middle|\; \begin{array}{l} S = \{s_1, \ldots, s_n\} \subset \mathbb{N}, t \in \mathbb{N}, \text{ so dass ein} \\ T \subseteq S \text{ existiert mit } \sum\limits_{x \in T} x = t \end{array} \right\}$$

Wir haben in der Vorlesung gesehen, dass SubsetSum NP-vollständig ist. Zeigen Sie, dass auch Benteler NP-vollständig ist.

**Aufgabe 31:** Bei der Sprache $0,1$-Programmierung sind $m$ Ungleichungen der Form

$$\sum_{i=1}^{n} a_{ij} x_i \leq d_j \text{ oder } \sum_{i=1}^{n} a_{ij} x_i \geq d_j,\ a_{ij} \in \mathbb{Z},\ d_j \in \mathbb{Z},\ j = 1, \ldots, m$$

gegeben. Zu entscheiden ist, ob eine Belegung der Variablen $x_i$ mit Werten aus $\{0, 1\}$ existiert, die alle Ungleichungen erfüllt. Zeigen Sie, dass $0,1$-Programmierung NP-vollständig ist. Tipp: Betrachten Sie dazu wieder die Sprache SubsetSum.
e eof small