Übungsblatt 12
(für die 28. Kalenderwoche 2008)

zur Vorlesung von Prof. Dr. J. Dassow
im Sommersemester 2008

Magdeburg, 1. Juli 2008

1. Gegeben ist folgendes Problem:

   Bin Packing (*BP*):

   Gegeben: $m \in \mathbb{N}$, $m > 0$, $s_i \in \mathbb{Q}$ mit $0 \leq s_i \leq 1$ für $1 \leq i \leq m$, $B \in \mathbb{N}$ mit $B \geq 1$.
   Frage: Existiert eine Zerlegung von $\{1, \ldots, m\}$ in Klassen I_k, $1 \leq k \leq B$, so dass $\sum_{i \in I_k} s_i \leq 1$ für alle k, $1 \leq k \leq B$, gilt?

   Beweisen Sie, dass das Problem *BP* $\mathbb{NP}$-vollständig ist, Sie dürfen dabei benutzen, dass die Probleme *SOS*, *SP* und *Knapsack* (siehe Übungsblatt 11) $\mathbb{NP}$-vollständig sind.

2. Bestimmen Sie für die folgenden *Bin Packing* Optimierungsprobleme mit $m > 0$ und den Werten

   a) $n > 0$, $m = 2n$ und

   $$s_i = \begin{cases} \frac{1}{2n} & \text{für } i \text{ gerade,} \\ \frac{1}{2} & \text{sonst,} \end{cases}$$

   b) $n > 0$, $m = 30n$, kleines $\varepsilon > 0$ und

   $$s_i = \begin{cases} \frac{1}{2} + \varepsilon & \text{für } i = 1, \ldots, 6n, \\ \frac{1}{4} - 2\varepsilon & \text{für } i = 6n+1, \ldots, 12n, \\ \frac{1}{4} + \varepsilon & \text{für } i = 12n+1, \ldots, 18n, \\ \frac{1}{4} - 2\varepsilon & \text{für } i = 18n+1, \ldots, 24n, \\ \frac{1}{4} + 2\varepsilon & \text{für } i = 24n+1, \ldots, 30n \end{cases}$$

   die Lösung nach *First-Fit*, die Lösung nach *Next-Fit*, die Lösung nach *First-Fit-Decreasing* sowie die optimale Lösung.

3. Beweisen Sie folgenden Satz:

   Satz: Falls $\mathbb{P} \neq \mathbb{NP}$, dann gibt es für das *Bin Packing* Optimierungsproblem keinen polynomiellen r-Approximationsalgorithmus[1] mit $r < \frac{3}{2}$.

   Hinweis: Führen Sie eine "Reduktion" des *Set Partition* Entscheidungsproblems auf die Existenz eines solchen Approximationsalgorithmus durch.

4. Zeigen Sie, dass der folgende Algorithmus ein 2-Approximationsalgorithmus für das *Vertex Cover* Problem ist:

   Eingabe: ungerichteter Graph $G = (V, E)$
   Ausgabe: Knotenüberdeckung C

   $C := \emptyset$;
   $E' := E$;
   while $E' \neq \emptyset$ **begin**
       $\{u, v\} :=$ beliebige Kante aus E';
       $C := C \cup \{u, v\}$;
       Lösche aus E' alle Kanten, die zu u oder v inzident sind;
   end;
   return C;

[1] Ein Algorithmus A heißt *r-Approximationsalgorithmus*, falls für alle Instanzen x des betrachteten Minimierungsproblems $\frac{m_A(x)}{m^*(x)} \leq r$ ist, wobei $m_A(x)$ die Approximationslösung und $m^*(x)$ die optimale Lösung der Instanz x sind.