

Übungsblatt 2

Aufgabe 2.1

6 Punkte

Um einen minimalen Spannbaum zu berechnen, geht der *Algorithmus von Prim* wie folgt vor: Ausgehend von einem beliebigen Knoten wird schrittweise ein Baum aufgebaut, indem jeweils die kürzeste Kante hinzugefügt wird, die den bisherigen Baum mit einem noch nicht zum Baum gehörenden Knoten verbindet.

Ein ähnliches Konzept verfolgt die Heuristik *Nearest Insertion* zur Berechnung einer minimalen Tour für das metrische TSP. Beginnend mit einem beliebigen Knoten, der eine Tour der Länge 0 darstellt, wird schrittweise eine größere Tour aufgebaut. Dazu wählt man unter den bisher nicht gewählten Knoten den, der einem bereits für die Tour gewählten Knoten am nächsten liegt, und fügt ihn davor oder dahinter in die Tour ein.

Zeigen Sie, dass die Heuristik *Nearest Insertion* eine 2-Approximation für das metrische TSP berechnet.

Aufgabe 2.2

4 Punkte

Wir nehmen an, dass $\mathcal{P} \neq \mathcal{NP}$ gilt. Zeigen oder widerlegen Sie die folgende Aussage: Es gibt ein Approximationschema für das Rucksackproblem mit Laufzeit $O(n^5 \cdot \log(\frac{1}{\varepsilon}))$.

Aufgabe 2.3

6 Punkte

Wir betrachten die *Entscheidungsvariante* von PARTITION: Für gegebene natürliche Zahlen b_1, \dots, b_n soll entschieden werden, ob es eine Teilmenge I_1 der Indexmenge $I = \{1, \dots, n\}$ gibt mit $\sum_{i \in I_1} b_i = \sum_{i \in I_2} b_i$, wobei $I_2 = I \setminus I_1$.

Geben Sie einen Algorithmus mit Laufzeit $O(n \cdot B)$ an, der die Entscheidungsvariante von PARTITION löst. Dabei sei $B = \sum_{i \in I} b_i$.

Aufgabe 2.4

2+6 Punkte

Bei den Problemen PARTITION DIFFERENCE und LOAD BALANCING geht es darum, für gegebene natürliche Zahlen b_1, \dots, b_n eine Teilmenge I_1 der Indexmenge $I = \{1, \dots, n\}$ zu finden, für die $|\sigma_1 - \sigma_2|$ bzw. $\max\{\sigma_1, \sigma_2\}$ minimal wird, wobei $\sigma_1 = \sum_{i \in I_1} b_i$ und $\sigma_2 = \sum_{i \in I_2} b_i$ sind.

Wir nehmen an, dass $\mathcal{P} \neq \mathcal{NP}$ gilt. Zeigen oder widerlegen Sie die folgenden Aussagen:

- Es gibt einen 2-Approximationsalgorithmus für PARTITION DIFFERENCE.
- Es gibt ein FPTAS für LOAD BALANCING.