10. Übungsblatt	Grundlagen der Theoretischen Informatik III	SoSe 2010
	Abgabe: bis 30.6.2010, 14:00 im Zettelkasten vor Raum H426	

1. Aufgabe: *(5 Punkte)*

Das Vertex-Cover Problem lässt sich auf zwei Arten formulieren, als funktionales oder als Entscheidungsproblem.

1. funktionales Problem: Finde ein k-elementiges Vertex-Cover zu G, falls ein solches existiert.
2. Entscheidungsproblem: Hat G ein k-elementiges Vertex-Cover?

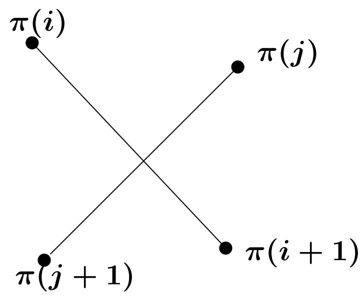
Zeigen Sie, dass die beiden Probleme aufeinander reduzierbar sind.

2. Aufgabe: *(7 Punkte)*

Wie Sie aus der Vorlesung wissen, ist das Traveling-Salesman Problem $\frac{2}{3}$-approximierbar, wenn auf der Distanzmatrix $M = (m_{ij})$ die Dreiecksungleichung gilt, d.h.:

$$\forall i, j, k : m_{ik} \leq m_{ij} + m_{jk}.$$

1. Bleibt die Dreiecksungleichung erhalten, wenn man alle Zeilen und Spalten um ihr Minimum reduziert?
2. Bei einem *Kugel-TSP* entspricht jeder Ort einem Punkt auf einer (fixen) Kugeloberfläche; die Distanz zwischen zwei Orten ist die Länge des kürzesten Bogens auf der Oberfläche, der die entsprechenden Punkte verbindet.
   a) Zeigen Sie, dass eine optimale Lösung eines Kugel-TSPs $\frac{2}{3}$-approximierbar ist.
   b) Zeigen Sie, dass eine optimale Lösung π eines Kugel-TSPs *kreuzungsfrei* ist, d.h. dass π keine Teilfolgen $\pi(i)\pi(i+1)$ und $\pi(j)\pi(j+1)$ enthält wie folgt



3. Aufgabe: (*8 Punkte)*

Eine Variante des RUCKSACK-Problems (vgl. 7. Übungsblatt) ist RUCKSACK MIT EINDEUTIGEN GEWICHTEN, kurz R_1, bei dem keine zwei Objekte dasselbe Gewicht haben. Eine knappe Formulierung davon ist

Problem: R_1

- Gegeben: $G = \{g_1, g_2, \ldots, g_n\} \subseteq \mathbb{N}$, $k \in \mathbb{N}$
- Gesucht: $S \subseteq G$ so dass $\sum_{s \in S} s$ maximal ist unter allen $S' \subseteq G$ mit $\sum_{s \in S'} s \leq k$.

Betrachten Sie den Algorithmus 'Stumpfpack'

Algorithmus 1 : Stumpfpack

```
Eingabe : G = {g_1, g_2, ..., g_n}, k ∈ ℕ
Ausgabe : S ⊆ G mit T = ∑_{s∈S} s ≤ k
S ← ∅
T ← 0
while i ≤ n do
    if T + g_i ≤ k then
        T ← T + g_i
        S ← S ∪ {g_i}
    i ← i + 1
```

1. Zeigen Sie, dass Stumpfpack *keine* $\frac{1}{2}$-Approximation an R_1 ist. Finden Sie dazu eine Eingabe, für die Stumpfpack ein T ausgibt, dass kleiner als die Hälfte einer optimalen Lösung ist.
2. Finden Sie eine Modifikation von Stumpfpack, die eine $\frac{1}{2}$-Approximation an R_1 ergibt. Der modifizierte Algorihtmus soll in $\mathcal{O}(n \log n)$ laufen.