

6. Übung:

Algorithmen und Komplexität

Wintersemester 2008

5. Januar 2009

Abgabe bis Montag, 12. Januar 2009, vor der Übung

Aufgabe 6.1: (5 Punkte)

Konstruieren Sie ein Single-Source-Shortest-Paths Problem auf dem Grundalgorithmus aus der Vorlesung (Auswahl und Behandlung einer beliebigen Kante, die die Dreiecksungleichung verletzt) möglichst viele Änderungen von DIST-Werten ausführt. *Hinweis:* Es sind 2^n Änderungen möglich.

Aufgabe 6.2: (5 Punkte)

Ein *längster Pfad* in einem *ungerichteten Baum* T ist ein Pfad in T mit maximaler Anzahl von Kanten. Zeigen Sie, daß folgender Algorithmus einen längsten Pfad berechnet:

- Wähle einen beliebigen Knoten u
- Verwende einen Suchalgorithmus, um einen Knoten v mit maximaler Entfernung von u zu finden.
- Verwende einen Suchalgorithmus, um einen Knoten w mit maximaler Entfernung von v zu finden.
- Der Pfad zwischen v und w ist ein längster Pfad in T .

Aufgabe 6.3: (15 Punkte)

Eine Euler-Tour ist ein Kreis, der jede Kante des Graphen genau einmal benutzt.

- Zeigen Sie, dass ein ungerichteter Graph ohne isolierte Knoten keine Euler-Tour besitzt, wenn er nicht zusammenhängend ist. (2 Punkte)
- Zeigen Sie, dass ein ungerichteter Graph keine Euler-Tour besitzt, wenn es einen Knoten mit ungeradem Grad gibt. (4 Punkte)

- c) Zeigen Sie, dass jeder zusammenhängende Graph bei dem jeder Knoten geraden Grad hat eine Euler-Tour besitzt. Hinweis: Entwerfen Sie einen Algorithmus, der eine Euler-Tour konstruiert und zeigen Sie, dass er unter den gegebenen Umständen immer funktioniert. Eine Möglichkeit besteht darin, sukzessive Kreise zu bestimmen und aneinanderzuhängen. (5 Punkte)
- d) Erklären Sie, wie Ihr Algorithmus aus c) in Zeit $O(m + n)$ implementiert werden kann. (4 Punkte)