

# Theoretische Informatik I

## 12. Übung

Geben Sie die Lösung der Aufgabe 3 bitte bis zum *20.01.2014 9:00 Uhr* ab. (Briefkasten vorm Raum 1/266 oder per eMail an [fal@informatik.tu-chemnitz.de](mailto:fal@informatik.tu-chemnitz.de), *Betreff: TI1 Hausaufgaben*)

**1. Aufgabe:** Ein *Eulerscher Kreis* in einem ungerichteten Graphen ist ein geschlossener Weg, in dem jede Kante des Graphen genau einmal vorkommt. Ein ungerichteter Graph  $G$  hat genau dann einen Eulerkreis, wenn  $G$  zusammenhängend ist und alle Knoten einen geraden Grad haben.

- (a) Konstruieren Sie aus dem Beweis dieser Aussage einen Algorithmus, der zu einem gegebenen Graphen  $G = (V, E)$  einen Eulerkreis ausgibt, falls ein solcher in  $G$  existiert.
- (b) Um die Laufzeit  $O(|V|+|E|)$  zu erreichen, muß der Algorithmus eine gefundene Kante in  $O(1)$  aus dem Graphen löschen können. Warum ist dies mit der herkömmlichen Adjazenzlistendarstellung kaum möglich?
- (c) Entwickeln Sie die Adjazenzliste zu einer Datenstruktur weiter, die es ermöglicht, den Algorithmus mit Laufzeit  $O(|V| + |E|)$  zu implementieren.
- (d) Geben Sie ein Verfahren an, wie die gegebene Adjazenzliste des Graphen in Ihre Datenstruktur umgewandelt werden kann. Beachten Sie, dass dafür nur Zeit  $O(|V|+|E|)$  zur Verfügung steht.

**2. Aufgabe:** Was ist  $64^{\log_4 n}$ ?

Zeigen Sie für  $a, b, c > 0$  und  $a, b, c \neq 1$

- (a)  $\log_a b \cdot \log_b c = \log_a c$ ,
- (b)  $\log_a b = \frac{1}{\log_b a}$ ,
- (c)  $c^{\log_b a} = a^{\log_b c}$ .

**3. Aufgabe:** Wir betrachten eine rekursiv implementierte binäre Suche auf einem sortierten Feld der Länge  $n$ . Der Einfachheit halber nehmen wir an, dass  $n$  eine Zweierpotenz ist.

- (a) Geben Sie eine *rekursive Implementierung* für die binäre Suche an.
- (b) Konstruieren Sie ein *worst-case-Beispiel* für  $n = 16$ . Geben Sie den Prozeduraufbaum für Ihr Beispiel an.
- (c) Wie groß ist die *worst-case-Laufzeit* der binären Suche allgemein? Zeigen Sie dies mittels Induktion.