Prof. Dr. A. Goerdt
Dr. L. Falke

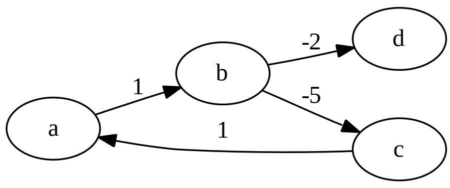
TU Chemnitz
Wintersemester 2017/2018
26.01.2018

Theoretische Informatik I

13. Übung

1. Aufgabe: Wir betrachten gerichtete, gewichtete Graphen, die auch Kreise negativer Länge enthalten können. Wir wollen kürzeste *einfache* Wege in solchen Graphen bestimmen.

(a) Betrachten Sie den folgenden Graphen und bestimmen Sie mit Hilfe dynamischer Programmierung den *kürzesten Weg* von a zu d sowie dessen *Länge*.

$G \quad =$ 

(b) Geben Sie den Aufrufbaum und die Rückgabewerte des rekursiven Algorithmus für den Aufruf `KW(`G, a, d`)` an.

(c) Geben Sie die Tabelle T an, wie sie bei der dynamischen Programmierung ausgefüllt wird.

2. 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.