

Aufgabe 1: (Breiten und Tiefensuche)

(6 Punkte)

Tiefen- und Breitensuche können auch auf ungerichteten Graphen ausgeführt werden. Dabei wird jede ungerichtete Kante so behandelt, als ob gerichtete Kanten in beide Richtungen vorhanden wären. Der Tiefen- beziehungsweise Breitensuchbaum ist definiert als der Teilgraph, der alle Baumkanten enthält.

Sei $G = (V, E)$ ein ungerichteter, zusammenhängender Graph und $u \in V$ beliebig aber fest. Mit $T_{DFS}(u)$, beziehungsweise $T_{BFS}(u)$ bezeichnen wir den Tiefensuchbaum und Breitensuchbaum von G , mit Wurzel u . Das heißt u war der Startknoten für die Breiten- bzw. Tiefensuche in G .

Zeigen Sie, dass falls $T_{DFS}(u) = T_{BFS}(u)$ gilt, dann gilt auch $G = T_{DFS}(u)$.

Aufgabe 2: (Algorithmenentwurf)

(6 Punkte)

Gegeben sei ein ungerichteter, zusammenhängender Graph $G = (V, E)$ in Adjazenzmatrixdarstellung. Geben Sie einen Pseudocode-Algorithmus der bestimmt, ob eine Kante $e \in E$ eine Schnittkante in G ist. Eine Kante $e \in E$ heißt *Schnittkante*, wenn nach Entfernen von e der Graph in zwei Zusammenhangskomponenten zerfällt.

Aufgabe 3: (AVL-Baum)

(6 Punkte)

Gegeben sei ein AVL-Baum T über eine Menge $S \subset \mathbb{N}$. Geben Sie ein Verfahren an, das T in zwei AVL-Bäume T_{even} und T_{odd} aufteilt, die nur gerade beziehungsweise ungerade Elemente aus S enthalten.

Aufgabe 4: (Rekursion)

(6 Punkte)

Betrachten Sie folgende Rekursionsgleichung: Ist n eine Zweierpotenz, so ist

$$\begin{aligned}T(n) &= \frac{n}{2} \cdot T(n/2) \\ T(1) &= 1\end{aligned}$$

Geben Sie eine geschlossene Form für $T(n)$ und benutzen dabei **nicht** das Mastertheorem. Beweisen Sie die Korrektheit Ihrer Antwort.

Aufgabe 5: (Dynamisches Programmieren)

(6 Punkte)

Gegeben sei eine Liste G von Münzbeträgen und einen Betrag b den Sie mit den Münzen in G bezahlen wollen. Dabei können gleiche Münzbeträge mehrfach in G enthalten sein. Sie wollen möglichst wenige Münzen verwenden, um b zu bezahlen. Geben Sie einen Algorithmus der das Problem für eine feste Liste G und einen Betrag b mittels Dynamischer Programmierung löst.