Übungsblatt 6

36.) Der Durchmesser eines Graphen ist die maximale Distanz zwischen zwei Knoten. Entwerfen Sie einen effizienten Algorithmus zur Bestimmung des Durchmessers eines Baumes und analysieren Sie dessen Laufzeit.

37.) Wir führen eine Tiefensuche für den Graphen $G = (V, E)$ durch. Zeigen Sie, dass die Kante (u, v) im Tiefensuchwald

- genau dann eine Baum- oder Vorwärtskante ist, wenn $d(u) < d(v) < f(v) < f(u)$ gilt,
- genau dann eine Rückwärtskante ist, wenn $d(v) \leq d(u) < f(u) \leq f(v)$ gilt,
- genau dann eine Querkante ist, wenn $d(v) < f(v) < d(u) < f(u)$ gilt.

38.) Schreiben Sie DFS und DFS-Visit so um, dass ein Stack verwendet wird und keine rekursiven Aufrufe mehr vorkommen. Ein Stack ist eine Liste S zusammen mit zwei Prozeduren Push(S, x) und Pop(S), wobei Push(S, x) das Element x am Ende der Liste anhängt und Pop(S) das letzte Element der Liste löscht und ausgibt.

39.) Vermutung: Falls ein gerichteter Graph einen Weg von u nach v besitzt und nach einer Tiefensuche $d(u) < d(v)$ gilt, dann ist v ein Nachfolger von u im Tiefensuchwald. Beweisen oder widerlegen Sie diese Vermutung!

40.) Gegeben sei ein gerichteter Graph und zwei seiner Knoten, u und v. Folgt aus der Existenz eines Weges von u nach v, dass nach einer Tiefensuche immer $d(v) < f(u)$ gilt?

41.) Modifizieren Sie die Prozeduren für die Tiefensuche so, dass bei Eingabe eines gerichteten Graphen alle Kanten zusammen mit ihrem Kantentyp ausgegeben werden. Sind weitere Modifikationen erforderlich, wenn die selbe Aufgabe für ungerichtete Graphen angewendet werden soll?

42.) Modifizieren Sie die Prozeduren für die Tiefensuche so, dass bei Eingabe eines gerichteten Graphen $V = (V, E)$ für jeden Knoten v ein weiteres Attribut $K(v)$ bestimmt wird, das die folgenden beiden Bedingungen erfüllt: (i) $K(v) \in \{1, 2, \ldots, k\}$, wobei k die Anzahl der Zusammenhangskomponenten von G ist, (ii) $K(u) = K(v)$ genau dann, wenn u und v in der selben Zusammenhangskomponente liegen.

43.) Ein gerichteter Graph $G = (V, E)$ heiße einzelzusammenhängend, wenn für jedes Knotenpaar $(u, v) \in V \times V$ höchstens ein Weg von u nach v existiert. Entwerfen Sie einen effizienten Algorithmus zum Feststellen, ob ein gerichteter Eingabegraph einzelzusammenhängend ist oder nicht.