

3. Übung:

Algorithmen und Komplexität

Wintersemester 2011

7. November 2011

Abgabe bis Montag, 14. November 2011 10:00 Uhr

Aufgabe 3.1: (10 Punkte)

Ein Graph $G = (V, E)$ heißt *bipartit*, wenn man V in zwei disjunkte Teilmengen A und B zerlegen kann, so daß $E \subseteq A \times B \cup B \times A$, d.h. jede Kante führt von einem Knoten aus A zu einem Knoten aus B oder umgekehrt.

Entwickeln Sie einen effizienten Algorithmus, der testet, ob ein gegebener Graph bipartit ist.

Hinweis: Verwenden Sie BFS (Breitensuche).

Aufgabe 3.2: (10 Punkte)

a) Zeigen Sie, dass ein Graph genau dann bipartit ist, wenn er keinen Kreis ungerader Länge enthält.

b) Erweitern Sie ihren Algorithmus aus Aufgabe 3.1 so, dass er im positiven Fall entsprechende Knotenmengen A und B berechnet und im negativen Fall einen Kreis ungerader Länge liefert (als Beweis für die Korrektheit des Ergebnisses).

Aufgabe 3.3: (10 Punkte)

Schreiben Sie eine Variante der Breitensuche $BFS(s)$, die

a) eine Einteilung des Graphen in Schichten berechnet, d.h. für jeden Knoten v eine ganze Zahl $layer[v]$ mit $layer[v]$ ist gleich der Länge eines kürzesten Pfades von s nach v , wobei die Länge eines Pfades P gleich der Anzahl der Kanten von P ist.

b) testet, ob der Graph azyklisch ist und im negativen Fall einen Kreis ausgibt.