

11. Übung:

Algorithmen und Komplexität

Wintersemester 2012-2013

1. Februar 2013

Wiederholungsübung

Aufgabe 11.1:

Sei $G = (V, E)$ ein gerichteter Graph, betrachten Sie folgenden Algorithmus.

- Berechne eine completion-Nummerierung (*compnum*) von G mit DFS.
- Drehe alle Kanten in G um
- Markiere alle Knoten als nicht besucht.
- Durchlaufe die Knoten in absteigender *compnum*-Reihenfolge und rufe für jeden noch nicht besuchten Knoten v die rekursive Funktion ($dfs(v)$) auf.

Zeigen Sie, dass jeder Aufruf von dfs in Zeile 4 des Algorithmus genau eine starke Zusammenhangskomponente von G durchläuft.

Aufgabe 11.2:

Sei $G = (V, E)$ ein gerichteter Graph und s und t zwei Knoten. Entwickeln Sie einen Algorithmus der eine maximale Zahl von knotendisjunkten Pfaden von s nach t findet.

Aufgabe 11.3:

Modifizieren Sie Dijkstra's Algorithmus so, daß er für einen gegebene Knoten t für alle $v \in V - \{t\}$ den kürzesten Pfad von v nach t berechnet.

Aufgabe 11.4:

Verwende das *orientation*-Prädikat, um zu testen, ob ein Punkt im Innern eines konvexen Polygons liegt. Wenn man diesen Test für viele Punkte immer mit demselben Polygon macht, kann man das ausnutzen, um einen einzelnen Test zu beschleunigen ?