

4. Übung:

Softwarepraktikum C++

Sommersemester 2009

12. Mai 2009

Abgabe per Email bis Mittwoch, 20. Mai 2009

Aufgabe 4.1:

(Punkte 10)

Implementierung/Visualisierung Tiefensuche

Berechnen Sie die DFS und die Completion Nummerierung eines Graphen $G = (V, E)$. Falls der Graph azyklisch ist (d.h. es existieren keine Rueckwaerts-kanten), berechnen Sie aus der Completion Nummer die topologische Sortierung und beschriften die Knoten. Ansonsten markieren Sie die Rueckwaerts-kanten und beschriften die Knoten mit DFS und Completionnummer. Eine topologische Sortierung ist eine injektive Funktion $topsort : V \rightarrow 1..n$ fuer $|V| = n$, so dass gilt $\forall (v, w) \in E$ ist $topsort(v) < topsort(w)$ Hinweise:

- Benutzen Sie GraphWin.
- http://www.algorithmic-solutions.info/leda_manual/graph.html.
- <http://www.leda-tutorial.org/en/unofficial/ch05.html>.

Aufgabe 4.2:

(Punkte 10)

Implementierung/Visualisierung bipartiter Graphen

Definition: Ein einfacher Graph $G = (V, E)$ heit bipartit, falls sich seine Knoten in zwei nicht leere, disjunkte Teilmengen A und B aufteilen lassen, sd. zwischen den Knoten innerhalb beider Teilmengen keine Kanten verlaufen. Das heit fuer eine Kante $v, w \in E$ gilt entweder $v \in A$ und $w \in B$ oder aber $w \in A$ und $v \in B$.

Implementieren Sie den Test mit Hilfe der **Breitensuche**. Faerben Sie die Knoten der Mengen A und B mit 2 Farben ein. Falls der Graph nicht bipartit ist, geben Sie eine Meldung aus.

Hinweise:

- Benutzen Sie GraphWin.
- Der Graph, den GraphWin zur Verfuegung stellt, ist zunaechst gerichtet. Simulieren Sie ungerichtete Graphen indem Sie nicht nur ueber ausgehende sondern auch ueber eingehende Kanten iterieren.
- http://www.algorithmic-solutions.info/leda_manual/graph.html.
- <http://www.leda-tutorial.org/en/unofficial/ch05.html>.