RWTH Aachen
Lehrgebiet Theoretische Informatik
Kneis—Langer—Rossmanith

WS 2009
Aufgabenblatt 4
23. November 2009

Softwareprojektpraktikum Graphalgorithmen

Aufgabe 15

Ändern Sie Ihre Graphbibliothek so, daß `tiefenSuche()` drei Arrays zurückgibt: Jeweils ein Array für Discovery- und Finish-Zeiten der einzelnen Knoten, und ein Array, welches die Baumkanten enthält. Verwenden Sie beispielsweise `out`-Parameter.

Zusätzlich kann optional eine weitere Funktion eine Datenstruktur `Tiefensuchwald` zurückgeben, welche man ähnlich zum `Tiefensuchbaum` nach Start- und Finish-Zeiten der Knoten und den Baumkanten befragen kann, ohne aber wissen zu müssen, in welchem `Tiefensuchbaum` sich ein Knoten tatsächlich versteckt (falls es mehrere Zusammenhangskomponenten gibt).

Aufgabe 16

Implementieren Sie den Test auf Kreisfreiheit neu und richtig, indem Sie die Tiefensuche nur einmal aufrufen (Hinweis: backedges).

Aufgabe 17

Wir erweitern das I/O-Format nun um Kantengewichte:

```
LuFGTI-Graph 1.0
UGraph|DiGraph
<Anzahl Knoten, numeriert von 0 - (n-1)>
<Anzahl nun folgender Kanten, numeriert von 0 - (m-1)>
<pro Zeile eine Kante "u v">
<Optionaler Teil (mit Schluesselwoertern eroeffnete Bloecke)>
EdgeWeights
<Pro Zeile je ein Gewicht der Kante mit den Nummern 0 - (m-1)>
<alle weiteren unbekannten Zeilen ignorieren>
```

Den Konstruktoren für gerichtete und ungerichtete Graphen soll man einen Dateinamen (und/oder `stream`) einer Datei im `.ltigraph`-Format übergeben können, aus denen der Graph dann erstellt wird.

Zusätzlich sollte es eine `read_graph()`-Funktion geben, welche eine gegebene Datei (oder einen gegebenen `stream`) einliest und einen (abstrakten) Graphen zurückgibt (je nachdem, ob die Datei einen gerichteten oder ungerichteten Graphen enthält).

Aufgabe 18

Implementieren Sie den "Bellman–Ford"-Algorithmus für kürzeste Wege:

Gegeben sei ein (gerichteter oder ungerichteter) Graph G mit Kantengewichten und einer seiner Knoten s. Gesucht ist die Distanz jedes Knotens von s. Ebenso soll es möglich sein, aus den Rückgabewerten des Algorithmus einen kürzesten Weg von s zu einem anderen gegebenen Knoten ablesen zu können.

Denken Sie an die Fehlerbehandlung, falls der Graph Kreise mit negativen Gewichten enthält.

Schreiben Sie ein Programm, welches als Argument eine Datei in obigem Graphformat bekommt, und ausgehend vom Knoten 0 die kürzesten Wege und ihre Länge zu allen anderen Knoten ausgibt.

Denken Sie bei allen Aufgaben an Dokumentation, Beispielprogramme und Unittests!