Technische Universität München
Institut für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Dmytro Chibisov
Hanjo Täubig

WS 2006/07
Aufgabenblatt 7
4. Dezember 2006

Praktikum Algorithmen-Entwurf

(Abgabetermin: Montag, den 11.12.2006, $14.^{00}$ Uhr)

Kürzeste und längste Pfade in DAGs

Aufgabe 1 Azyklische Graphen

Gegeben sei ein gerichteter Graph $G = (V, E)$ mit beliebigen ganzzahligen Kantengewichten, der keine Kreise enthält. Implementieren und animieren Sie einen Algorithmus, der *in linearer Zeit*, also in Zeit $O(|V| + |E|)$ statt $O(|V| \log |V| + |E|)$ wie beim Algorithmus von Dijkstra, die kürzesten Pfade von einem vom Benutzer gewählten Startknoten v zu allen von v erreichbaren Knoten berechnet.

Erweitern Sie das Programm so, daß der Benutzer wählen kann, ob die kürzesten oder die längsten Pfade ermittelt werden sollen. In beiden Fällen sollen die Pfade in linearer Zeit berechnet werden.

Hinweise

Sie können den Algorithmus aus Aufgabe 1 mit den Graphen `acyc1.gw` und `acyc2.gw` testen. Die Gewichte der Kanten sind wieder als Strings im User-Label gespeichert und sollten in ein `edge_array<double>` eingelesen werden.

Beachten Sie, daß die bearbeiteten Graphen dieses Mal gerichtet sind. Ihre Programme sollten daher ein `gw.set_directed(true)` und *kein*
`g.make_undirected()` enthalten.

Fluß in Graphen

Aufgabe 2 Algorithmus von Goldberg und Tarjan

Gegeben ist ein gerichteter Graph $G = (V, E)$, in dem jeder Kante $e \in E$ eine ganzzahlige Kapazität $c(e) \in \mathbb{N}$ zugeordnet ist. In G sind eine Quelle $s \in V$ (*source*) und eine Senke

$t \in V$ (*target*) ausgezeichnet. Implementieren und animieren Sie den Algorithmus von Goldberg und Tarjan, der einen maximalen Fluß in dem gegebenen Graphen berechnet.

Der Benutzer soll vor Ablauf des Algorithmus Quelle und Senke auswählen. Anschließend soll die Arbeitsweise des Algorithmus gut am Bildschirm mitverfolgt werden können. Insbesondere sollten die aktiven Knoten gekennzeichnet und jede ausgeführte Push- und Relabel-Operation anschaulich animiert werden. Dabei ist es sinnvoll, bei jedem Knoten den Überschuß und den Distanz-Wert als Label anzuzeigen.

Hinweise

Als Eingabe für Ihren Algorithmus können Sie die gerichteten Graphen `flow1.gw` bis `flow4.gw` verwenden. Bei diesen Graphen sind die ganzzahligen Kapazitäten der Kanten als Strings im User-Label der Kanten gespeichert. Wird der linkeste Knoten als Quelle und der rechteste als Senke gewählt, ergeben sich für `flow1.gw` bis `flow4.gw` die folgenden Flußwerte: 40, 105, 340 und 50.