

Einführung in die Diskrete Mathematik

6. Übung

1. Man nenne einen s - t -Präfluss f maximal, wenn $\text{ex}_f(t)$ maximal ist.
 - (a) Man zeige, dass es für jeden maximalen Präfluss f einen maximalen Fluss f' mit $f'(e) \leq f(e)$ für alle $e \in E(G)$ gibt.
 - (b) Man zeige, wie man in $O(nm)$ Zeit einen maximalen Präfluss in einen maximalen Fluss umwandeln kann. (4 Punkte)
2. Man zeige, dass der PUSH-RELABEL-ALGORITHMUS $O(n^2m)$ nichtsaturierende Pushes durchführt, unabhängig von der Wahl des aktiven Knotens v in ③. (4 Punkte)
Hinweis: Betrachten Sie $\Phi := \sum_{v \text{ aktiv}} \psi(v)$.
3. Implementieren Sie den PUSH-RELABEL-ALGORITHMUS zur Berechnung eines maximalen s - t -Flusses. Ihr Programm sollte höchstens $O(n^2m)$ Laufzeit haben, idealerweise nur $O(n^2\sqrt{m})$. Ein genauere Beschreibung der Aufgabe finden Sie umseitig. (16 Punkte)

Abgabe:

Aufgaben 1 und 2: Dienstag, den 23.11.2010, **vor** der Vorlesung.

Aufgabe 3: Dienstag, 30.11.2010, **vor** der Vorlesung.

Hinweise zur Programmierübung:

Einlesen der Daten: Dem Programm muss beim Aufruf der Name einer Datei übergeben werden. Ein Aufruf hat also die Form

```
<programmname> <dateiname>
```

Eine gültige Datei, die einen Graphen beschreibt, hat das folgende Format:

Knotenanzahl

Kantenzahl

Knoten0a Knoten0b Kapazitaet0

Knoten1a Knoten1b Kapazitaet1

...

Die Einträge der Datei sind ausschließlich ganze Zahlen. Sie können voraussetzen, dass die Summe aller Zahlen in der Eingabe kleiner als 2^{31} ist. In den ersten beiden Zeilen steht jeweils eine einzelne natürliche Zahl, welche in der ersten Zeile die Anzahl der Knoten und in der zweiten die Anzahl der Kanten angibt. Jede weitere Zeile spezifiziert genau eine Kante. Die ersten beiden Einträge einer Zeile sind zwei verschiedene nichtnegative ganze Zahlen, welche die Nummern der Endknoten der Kante sind. Dabei nehmen wir an, dass, wenn wir n Knoten haben, die Knoten von 0 bis $n - 1$ durchnummeriert sind. Der dritte Eintrag in der Zeile ist eine positive ganze Zahl, die die Kapazität der Kante bezeichnet. Der Index einer jeden Kante ist durch ihre Zeilennummer in der Eingabedatei gegeben: Zeile i kodiert die Kante mit Index $i - 3$ (für $i = 3, \dots, m + 2$, wobei m die Zahl der Kanten sei). Die Kanten sind dadurch auch von 0 bis $m - 1$ durchnummeriert. Parallele Kanten können vorkommen.

Sie können voraussetzen, daß es mindestens 2 Knoten gibt. Der Knoten s hat dabei Nummer 0 und der Knoten t Nummer 1.

Ausgabeformat: Das Programm muss in der ersten Zeile der Ausgabe den Wert eines maximalen s - t -Flusses ausgeben. Die weiteren Zeilen müssen jeweils genau einen Index einer Kante und den zugehörigen Flusswert enthalten. Es werden dabei nur die Kanten mit positivem Fluss ausgegeben, und die Zeilen sollen nach dem Kantenindex aufsteigend sortiert sein.

Beispiel: Eine Eingabedatei für einen Graphen mit fünf Knoten und sechs Kanten kann so aussehen:

5

6

```
0 1 2
2 1 2
2 3 2
3 1 1
0 2 3
3 4 2
```

Die Ausgabe der Programms muss dann so aussehen:

```
5
0 2
1 2
2 1
3 1
4 3
```

Das Programm muss in C oder C++ geschrieben sein. Es muss korrekt arbeiten und ohne Fehlermeldung kompiliert werden können. Der Code muss auf einem gängigen Linuxsystem funktionieren. Für die Sortierung dürfen Sie `qsort` oder `std::sort` verwenden. Andere Algorithmen aus externen Bibliotheken dürfen nicht verwendet werden. Achten Sie auch darauf, dass Sie Ihr Programm ausreichend mit Kommentaren versehen. **Abgabe:** Der Quelltext der Programms muss bis **30. November, 16:15 Uhr** per E-Mail beim jeweiligen Tutor eingegangen sein. Außerdem ist bis zu diesem Zeitpunkt ein Ausdruck des Quelltextes zusammen mit den Theorieaufgaben abzugeben.

Testinstanzen befinden sich auf der Seite

http://www.or.uni-bonn.de/lectures/ws10/edm_uebung_ws10.html