Technische Universität München
Institut für Informatik
Prof. Dr. Angelika Steger

SS 2003
Aufgabenblatt 1
07.04.2003

Praktikum Diskrete Optimierung

Abgabetermin: Montag, den 14.04.2003, 14.00 Uhr

Aufgabe 1 First Steps

Machen Sie sich mit C++ und insbesondere mit der Verwendung der STL-Bibliothek von C++ sowie mit der Verwendung von Makefiles vertraut. Schreiben Sie sich ein einfaches C++-Programm sowie ein einfaches Makefile, übersetzen und testen Sie Ihr Programm auf unseren Praktikumsrechnern.
Makefile und Programmquelltext zu dieser Aufgabe müssen nicht abgegeben werden.

Aufgabe 2 Artikulationsknoten in ungerichteten Graphen

Im folgenden sei $G = (V, E)$ ein ungerichteter, zusammenhängender Graph.

Definition 1 *Der (Knoten-)Zusammenhang von G ist die minimale Anzahl von Knoten, deren Entfernung G in zwei oder mehr Zusammenhangskomponenten zerfallen läßt.*

Ein Graph mit Zusammenhang mindestens 2 heißt auch *zweifachzusammenhängend.*

Definition 2 *Ein Knoten v von G heißt Artikulationsknoten, wenn G durch Entfernung von v in zwei oder mehr Zusammenhangskomponenten zerfällt.*

Offensichtlich ist G genau dann zweifachzusammenhängend, wenn er keinen Artikulationsknoten enthält.
Die *Zweifachzusammenhangskomponenten* (auch *Blöcke* oder engl. *biconnected components*) von G sind die maximalen Teilgraphen von G, die zweifachzusammenhängend sind.
Der Zusammenhang von Graphen und die Berechnung der Blöcke ist zum Beispiel beim Aufbau ausfallsicherer Kommunikationsnetzwerke interessant. Ein zweifachzusammenhängendes Teilnetzwerk bleibt auch dann zusammenhängend, wenn ein beliebiger Knoten ausfällt.

Problemstellung

Implementieren Sie die Suche nach Artikulationsknoten in ungerichteten Graphen. Artikulationsknoten können mit Hilfe einer modifizierten Tiefensuche gefunden werden (näheres hierzu finden Sie u.a. in der angegebenen Literatur). Ihr C++-Programm soll den Graphen von der Standardeingabe lesen und die Artikulationsknoten auf der Standardausgabe ausgeben.
Wichtiger Hinweis: Halten Sie das im folgenden angegebene Eingabe- und Ausgabeformat bitte genau (keinerlei zusätzliche Leerzeichen, Zeilenumbrüche etc.) ein! Die abgegebenen Programme werden u.a. auch mit Hilfe eines automatisierten Verfahrens auf verschiedenen Eingaben getestet.

Eingabeformat

Als Eingabe erhält Ihr Programm zunächst eine Zeile mit der Knotenanzahl n des Graphen. Die Knoten des Graphen werden durch Zahlen zwischen 0 und $n-1$ repräsentiert. In den darauffolgenden Zeilen werden die Kanten des Graphen als Paare von Knoten angegeben, wobei jede (ungerichtete) Kante in der Auflistung nur einmal vorkommt.

Beispieleingabe

```
6
0 1
0 2
2 3
3 4
3 5
4 5
1 2
```

Ausgabeformat

Ihr Programm soll die Artikulationsknoten des Eingabegraphen in einer beliebigen Reihenfolge ausgeben. Jeder Artikulationsknoten soll hierbei in einer eigenen Zeile ausgegeben werden, ebenso soll jeder Artikulationsknoten *genau einmal* ausgegeben werden.

Beispielausgabe

```
3
2
```

Weitere Beispieleingaben und -ausgaben stehen unter
`http://www14.in.tum.de/lehre/2003SS/optprak/data.html`
zur Verfügung.

Zeitlimit und Abgabe

Pro Testeingabe stehen maximal 20 Sek. Laufzeit zur Verfügung.
Die Abgabe der Lösung muß bis Montag, 14.04.2003, 14.^{00} Uhr, per E-Mail an die Adresse `optprak@in.tum.de` erfolgen.

Literaturhinweise

- Turau. Algorithmische Graphentheorie. Addison Wesley, 1996
- Sedgewick. Algorithms. Addison-Wesley, 1988

Praktikumsbetreuung:

Stephan Micklitz und Andreas Weißl, optprak@in.tum.de