

Einführung in die Diskrete Mathematik

2. Übung

1. Sei $G = (V, E)$ ein stark zusammenhängender gerichteter Graph mit n Knoten. Zeigen Sie, dass G dann einen stark zusammenhängenden Teilgraphen $G' = (V, E')$ mit $|E'| \leq 2n - 2$ enthält. (2 Punkte)
2. Beweisen Sie die folgenden Aussagen:
 - (a) Ein ungerichteter Graph ist genau dann 2-fach kantenzusammenhängend, wenn er mindestens zwei Knoten und eine Ohrenzerlegung hat.
 - (b) Ein gerichteter Graph ist genau dann stark zusammenhängend, wenn er eine Ohrenzerlegung hat.
 - (c) Die Kanten eines ungerichteten Graphen G mit mindestens zwei Knoten können genau dann so orientiert werden, dass der resultierende gerichtete Graph stark zusammenhängend ist, wenn G 2-fach kantenzusammenhängend ist. (2+2+2 Punkte)

Abgabe: Dienstag, den 29.10.2013, **vor** der Vorlesung.

3. Programmieraufgabe

Schreiben Sie ein Programm, das zu einem gegebenen gerichteten Graphen die starken Zusammenhangskomponenten bestimmt. Das Programm soll lineare Laufzeit haben. Es soll, im Unterschied zum Algorithmus, der in der Vorlesung vorgestellt wurde, keine Rekursion benutzen. (12 Punkte)

Zur Programmieraufgabe:

Zum Einlesen und Speichern der Graphen ist die Klasse `Graph` aus der Vorlesung "Algorithmische Mathematik I" aus dem Wintersemester 2012/2013 zu verwenden.

Alle Datenstrukturen, die in der Vorlesung "Algorithmische Mathematik I" vorgestellt wurden, finden Sie hier:

<http://www.or.uni-bonn.de/~vygen/lectures/alma1ws12.html>

Eingabeformat: In der ersten Zeile steht eine einzelne natürliche Zahl (größer als 0 und kleiner als 2^{31}), welche die Anzahl der Knoten angibt. Wir nehmen an, dass, wenn wir n Knoten haben, die

Knoten von 0 bis $n - 1$ durchnummeriert sind. Jede weitere Zeile spezifiziert genau eine Kante. Die beiden Einträge einer Zeile sind zwei verschiedene nichtnegative ganze Zahlen, welche die Nummern der Endknoten der Kante sind. Die Kanten sind jeweils vom ersten angegebenen Knoten zum zweiten gerichtet. Es können parallele Kanten, aber keine Schleifen vorkommen. Der Graph muss nicht zusammenhängend sein. Die Sortierung der Kanten in der Eingabedatei kann beliebig sein.

Ausgabeformat: Die erste Zeile der Ausgabe muss die Zahl der gefundenen starken Zusammenhangskomponenten enthalten. Jede weitere Zeile soll die Nummern der Knoten von einer starken Zusammenhangskomponente enthalten. Aufeinanderfolgende Einträge sollen durch einzelne Leerzeichen getrennt werden. In den einzelnen Zeilen sollen die Knotennummern aufsteigend sortiert sein.

Beispiel: Eine Eingabedatei für einen Graphen mit 4 Knoten und 5 Kanten kann so aussehen:

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

Die Ausgabe der Programms kann dann so aussehen (die zweite und die dritte Zeile können auch vertauscht sein):

```
2
0 1 3
2
```

Das Programm muss in C++ geschrieben sein. Es muss korrekt arbeiten und ohne Fehlermeldung kompiliert werden können. Der Code muss auf einem gängigen Linuxsystem funktionieren. Für die Sortierung der Ausgabe dürfen Sie `qsort` oder `std::sort` verwenden. Die Laufzeit dieser Sortier Routinen soll hier nicht betrachtet werden. Andere Algorithmen aus externen Bibliotheken dürfen nicht verwendet werden.

Abgabe: Der Quelltext der Programms muss bis 29. Oktober, 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 ab dem 23.10.2013 auf der Seite

http://www.or.uni-bonn.de/lectures/ws13/edm_13_uebung.html

Sollten weitere Hinweise zu der Programmierübung notwendig sein, werden diese ebenfalls auf dieser Homepage bekanntgegeben.