

# Algorithmische Graphentheorie für Informatiker

## Labor 2

### (Aufgabe 1)

- a) **(2 Punkte)** Verändern Sie die Klasse Graph. Implementieren Sie einen Konstruktor für diese Klasse mit einem einzigen Parameter, einen Dateinamen, der die Eingabedaten für einen ungerichteten Graphen enthält. Die Datei sollte die folgende Struktur haben: Die erste Zeile enthält zwei Ganzzahlen,  $n$  und  $m$ , wobei  $n$  die Anzahl der Knoten und  $m$  die Anzahl der Kanten darstellt. Die folgenden  $m$  Zeilen repräsentieren die Kanten (Knoten1 Knoten2), wobei jede Zeile zwei durch ein Leerzeichen getrennte Ganzzahlen enthält, die die verbundenen Knoten angeben. Die Adjazenzliste wird als ein Array von dynamischen Vektoren repräsentiert.

Die Methode `sole` die Daten aus der Datei lesen und sie in einer geeigneten Datenstruktur speichern, die den Graphen repräsentiert (Repräsentierung auf Adjazenzliste).

- a) **(2 Punkte)** Implementieren Sie die Methode `zeigeAngrenzendeKnoten`. Diese Methode akzeptiert als Parameter einen bestimmten Knoten im Graphen. Die Methode sollte alle an den gegebenen Knoten  $n$  angrenzenden Knoten identifizieren und auf dem Bildschirm anzeigen, zusammen mit dem Grad des Knoten  $n$ .

### (Aufgabe 2)

**(4 Punkte)** Implementieren Sie die Methode `printPfad(int x, int y)`, welche als Parameter zwei Knoten eines Graphen erhält. Die Methode soll alle Pfade der Länge 2 zwischen diesen beiden Knoten ausgeben. Falls kein solcher Pfad existiert, sollte die Methode die Nachricht "Does not exist" ausgeben.

### (Aufgabe 3)

**(1 Punkt)** Erklären Sie den implementierten Code und beantworten Sie die Fragen der Lehrkraft.

### BEMERKUNGEN

- **Arbeitszeit: 60 Minuten**
- **1 Punkt von Amts Wegen**
- **Im Falle von abgeschriebenem Code oder inkorrekt repräsentiert wird die Laborarbeit mit 1 benotet. Das heisst, man verwendet KEINE Adjazenzmatrix**

