

Algorithmen und Datenstrukturen

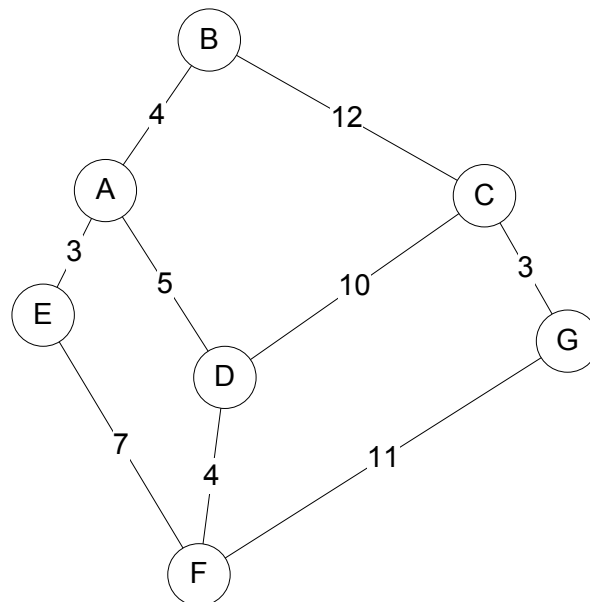
Sommersemester 2012

11. Übungsblatt

Hinweis: Bitte beachten Sie, dass die AuD-Vorlesung am Freitag, den 15. Juni 2012 entfällt. Zudem finden die Vorlesungen am 11.07.2012 bzw. 13.07.2012 im Hörsaal V1 bzw. V2 statt. Weitere Details können der Webseite zur Vorlesung entnommen werden.

Aufgabe 1: Dijkstra

Gegeben sei folgender ungerichteter, gewichteter Graph G :



Wenden Sie den Algorithmus von Dijkstra zur Ermittlung der kürzesten Wege ausgehend von Knoten A auf den Graphen an. Geben Sie für jeden Zwischenschritt den gewählten Knoten und die Randknoten jeweils mit Entfernung und Vorgänger an. (9 Punkte)

Aufgabe 2: A* - Implementierung

In dieser Aufgabe soll der A*-Algorithmus implementiert werden. Auf der Website zur Übung steht das Ausgangsprojekt zum Herunterladen bereit. Die Klasse `IfisNavi` verwaltet eine Liste von Städten (`City`). Jede Stadt verfügt über einen Namen, Positionsangaben in Form von Breiten- und Längengrad sowie über eine Reihe von Verbindungen (`Street`) zu anderen Städten. Eine Verbindung besitzt eine Zielstadt und ein Gewicht bzw. Länge. In der Klasse `City` wurden u.a. bereits die Methoden `equals(Object o)` und `hashCode()` implementiert, so dass diese Klasse zum Beispiel als Schlüssel einer `HashMap` genutzt werden kann.

Die Methode `loadMap()` konstruiert einen Graphen, welcher zum Testen der zu implementierenden Methode genutzt werden soll. Machen Sie sich mit dem Code vertraut bevor Sie mit der Implementierung beginnen. Nutzen Sie bei Ihnen unbekanntenen Klassen ggf. die Java API (<http://download.oracle.com/javase/6/docs/api/>).

Implementieren Sie die Methode

```
public List<City> aStarSearch(City from, City to)
```

zur Ermittlung des kürzesten Weges nach dem **A*-Algorithmus**. Die Methode soll die Liste der zu besuchenden Städte des kürzesten Weges zurückliefern. Die Klasse `City` stellt die Methode `linearDist(City c)` zur Berechnung der Luftlinien-Entfernung bereit.

Welche Strecke liefert Ihr Algorithmus von *Chorzow* nach *Kiew* bzw. von *Breslau* nach *Donezk*?
(11 Punkte)

Aufgabe 3: Kruskal

Berechnen Sie für den in Aufgabe 1 gegebenen Graphen G mittels des **Algorithmus von Kruskal** einen minimalen spannenden Baum. Geben Sie die Reihenfolge der betrachteten Kanten und die Entscheidung (*gewählt* oder *verworfen*) an. Zeichnen Sie den resultierenden Graph.
(5 Punkte)

Beachten Sie die Hinweise auf dem ersten Übungsblatt.

Abgabetermin: Donnerstag, den 21. Juni bis 10 Uhr im Institut für Informationssysteme, 2. OG im Informatik-Neubau (Raum 2.071, Küche neben Sekretariat)