

Übungsblatt 1

Aufgabe 1

Schicken Sie pro Team eine Email an `gaertler@ira.uka.de` mit Benutzernamen und Passwörtern (unverschlüsselt), um lokale CVS-Benutzer zu erhalten. Laden Sie anschließend die `yFiles` und `GraphML` aus dem CVS-Depot herunter. Zur Erinnerung:

- Secure tunnel:
`ssh -N -L 2401:i11raid.ira.uka.de:2402 ↯`
`algoprakt@i11raid.ira.uka.de`
- Depot: `/home/algo/lehre/cvsroot`

Aufgabe 2

- (a) Implementieren Sie eine Breitensuche. Benutzen Sie den folgenden Pseudo-Code als Richtlinie.

Eingabe : Graph $G = (V, E)$, Wurzel $s \in V$

Daten : Queue Q (für Knoten in BFS-Front)
Knoten- und Kantenmarkierungen

```
markiere  $s$ ;  
initialisiere  $Q$ ;  
füge an  $Q \leftarrow s$ ;  
→ root( $s$ )  
while  $Q$  nicht leer do  
  entferne  $v \leftarrow Q$ ;  
  foreach nicht markierte Kante  $e = \{v, w\} \in E$  do  
    markiere  $e$ ;  
    if  $w$  nicht markiert then  
      markiere  $w$ ;  
      füge an  $Q \leftarrow w$ ;  
      → traverse( $v, e, w$ )  
→ done( $s$ )
```

(b) Modifizieren Sie Ihre Klasse in folgender Weise:

- Ersetzen Sie die Knotenmarkierung durch **Integer** Werte, die den Abstand zur Wurzel wiedergeben.
- Fügen Sie eine Option ein, die den Graphen als gerichtet oder ungerichtet durchsucht.
- Fügen Sie vier Methoden **traverseTreeEdge**, **traverseForwardEdge**, **traverseBackwardEdge** und **traverseCrossEdge** ein, und modifizieren Sie **traverse** so, dass dort die entsprechende Methode aufgerufen wird. Zur Erinnerung: eine Kante $e = (u, v)$ heißt:
 - Baumkante, falls $\text{BFS}(v) = \infty$,
 - Vorwärtskante, falls $\text{BFS}(v) > \text{BFS}(u)$,
 - Rückwärtskante, falls $\text{BFS}(v) < \text{BFS}(u)$,
 - Querkante, falls $\text{BFS}(u) = \text{BFS}(v)$,

wobei **BFS** den aktuellen Abstand zur Wurzel angibt und $\text{BFS}(v) = \infty$, falls der Knoten v noch nicht markiert ist. (Zu beachten ist, dass diese Bedingungen abgeprüft werden, wenn die Kante markiert wird.)

Aufgabe 3

Schreiben Sie eine Layout-Klasse für Bäume (mit ausgewählter Wurzel), die Knoten über ihren Vorgängern geschickt platziert. Benutzen Sie eine Breitensuche, um zu testen, ob der Graph wirklich ein Baum ist.

Hinweise: Merken Sie sich für jede Teilwurzel, das Intervall, in dem sie platziert werden darf, und teilen Sie den Kindknoten ein Intervall zu, das proportional ist zu seiner Teilbaumgröße.