

## Übungsblatt 10

Abgabe bis Dienstag, den 2<sup>1.</sup> Juli um 2<sup>4:</sup>00 Uhr

### Aufgabe 1 (10 Punkte)

Sei  $G = (V, E)$  ein beliebiger gerichteter Graph, und sei für zwei Knoten  $u, v \in V$  die Entfernung  $\text{dist}(u, v)$  definiert als die Länge des kürzesten Pfades (gemessen in der Anzahl Kanten) von  $u$  nach  $v$ .

Beweisen Sie, dass nach einer Breitensuche (BFS) auf  $G$  mit Startknoten  $s$ , Level  $i$  genau die Knoten  $v \in V$  mit  $\text{dist}(s, v) = i$  enthält. (5 Punkte)

Überlegen Sie, ob man auch eine Tiefensuche (DFS) von  $s$  aus zur Berechnung aller  $\text{dist}(s, v)$  mit  $v \in V$  verwenden kann. Falls ja, zeigen Sie wie. Falls nein, begründen Sie stichhaltig warum nicht. (5 Punkte)

*Hinweis:* Für den Beweis für BFS bietet sich vollständige Induktion über  $i$  an. Für die Frage zu DFS ist der ontologische Gottesbeweis von Gödel nicht hilfreich, auch wenn es erst mal so aussieht.

### Aufgabe 2 (10 Punkte)

Implementieren Sie Dijkstra's Algorithmus gemäß der auf dem Wiki verlinkten Vorlage (aufgrund anhaltender überbordender Nettigkeit schon wieder sowohl in Java als auch in C++ vorhanden).

Der Unit Test ist bereits vorgegeben. Achten Sie darauf, dass Ihre Methode das im Unit Test Feld von Strings *dijkstraStates* richtig setzt. Sie können den Unit Test bei Bedarf erweitern, müssen es aber nicht.

Achten Sie insbesondere darauf, dass Sie den Algorithmus nicht länger laufen lassen wie nötig, also genau so lange, bis die Distanz zum gegebenen Zielknoten mit Sicherheit bekannt ist.

Committen Sie Ihren Code (für Aufgabe 2) + Beweis (für Aufgabe 1) in unser SVN, in einen neuen Unterordner *uebungsblatt\_10*. Committen Sie in diesem neuen Unterordner außerdem wie gehabt eine Textdatei *erfahrungen.txt*. Beschreiben Sie dort in ein paar Sätzen Ihre Erfahrungen mit diesem Übungsblatt und der Vorlesung dazu. Insbesondere: Wie lange haben Sie ungefähr gebraucht? An welchen Stellen gab es Probleme und wieviel Zeit hat Sie das gekostet? Wo steht aktuell der Frühlingspunkt, und wann wird er das nächste Mal wieder an der selben Stelle stehen,

und warum?