

Präsenzaufgaben 7

02.05.2023

Die Lösung der Aufgaben wird am Ende der Übung von Ihnen vorgestellt.

Aufgabe 1

Verwenden Sie die Klasse `Graph` von Präsenzblatt 6 erneut als Vorlage. Ergänzen Sie nun die folgenden Methoden:

```
public boolean isDirected()

public int[] getDistances(int start)
```

- `isDirected` gibt zurück, ob der Graph gerichtet (**true**) oder ungerichtet (**false**) ist. Tipp: Sie können die Adjazenzliste von oben nach unten durchlaufen und sich damit die Breiten- oder Tiefensuche ersparen.
- `getDistances` gibt ein Feld mit den Entfernungen aller Knoten zum Start-Knoten zurück:
 - Das Feldelement 0 (Dummy-Knoten) soll den Wert -2 erhalten.
 - Alle nicht erreichbaren Knoten erhalten den Wert -1.
 - Der Startknoten selbst erhält den Wert 0.
 - Alle anderen Feldelemente erhalten die Anzahl der Kanten der kürzesten Wegstrecke vom Startknoten zum entsprechenden Knoten.
 - Tipp: Verwenden Sie die Breitensuche bei Ihrem Algorithmus.

Aufgabe 2

Beschreiben Sie den Lösungsalgorithmus von Aufgabe 2 aus Vorlesung 12 (Folie 312, Vorlesungsvideo von 2019 ab Minute 19:40) in eigenen Worten. Hier die Aufgabenstellung:

- Studierende einer Hochschule wollen ein Benefiz-Rugby-Spiel organisieren. Aus Sorge vor Repressalien sollen Studenten nur gegen Professoren spielen, bei denen Sie keine Prüfung mehr ablegen müssen. Umgekehrt sollen, im Sinne der Teambildung, in einem Team nur Professoren sein, die allen Studenten ihres Teams noch mindestens eine Prüfung abnehmen müssen.
- Entwickeln Sie einen Algorithmus, der entscheidet, ob die Teilnehmer unter diesen Voraussetzungen grundsätzlich auf zwei Teams aufgeteilt werden können.
- Ihr Algorithmus sollte eine Komplexität von $O(n + r)$ haben, wobei n die Zahl der Teilnehmer und r die Anzahl der Paarungen Professor, Student ist, die sich in keiner Prüfung mehr begegnen werden.