Albert-Ludwigs-Universität
Institut für Informatik
Prof. Dr. Fabian Kuhn
P. Bamberger, Y. Maus, P. Schneider

Informatik 2 - Sommersemester 2018
Übungsblatt 7

Abgabe: Montag, 18. Juni, 14:00 Uhr

Aufgabe 1: Tiefensuche in ungerichteten Graphen *(7 Punkte)*

Sei $G = (V, E)$ ein Graph und $(u, v) \in E$ eine Kante. Wird während der Ausführung von `DFS-visit`(u) ein rekursiver Aufruf `DFS-visit`(v) gestartet, dann klassifizieren wir (u, v) wie folgt:

- v ist weiß $\Longrightarrow$ (u, v) ist eine Baumkante.
- v ist grau $\Longrightarrow$ (u, v) ist eine Rückwärtskante.
- v ist schwarz und v wurde *nach* u grau $\Longrightarrow$ (u, v) ist eine Vorwärtskante.
- v ist schwarz und v wurde *vor* u grau $\Longrightarrow$ (u, v) ist eine Querkante.

Zeigen Sie dass `DFS-visit`(s) in einem *ungerichteten* Graphen $G = (V, E)$ und $s \in V$ nur Baumkanten und Rückwärtskanten erzeugt.

Aufgabe 2: Graphen auf Kreisfreiheit testen *(6 Punkte)*

Sei $G = (V, E)$ ein ungerichteter Graph repräsentiert durch eine Adjazenzliste. Beschreiben Sie einen Algorithmus oder geben Sie Pseudocode an, mit dem man G in $\mathcal{O}(|V|)$ Schritten auf Kreisfreiheit testen kann.

Hinweis: Welche obere Schranke für die Anzahl der Kanten hat ein ungerichteter kreisfreier Graph?

Aufgabe 3: Minimaler Spannbaum *(7 Punkte)*

Gegeben sei ein gewichteter, ungerichteter, zusammenhängender Graph G. Beweisen Sie, dass folgender Algorithmus terminiert und einen minimalen Spannbaum von G berechnet:

Algorithm 1: `MST`$\big(G = (V, E, w)\big)$

```
while G has cycle C ⊆ E do
    Let e ∈ C with maximum weight.
    E ← E \ {e}
return G
```

Hinweise: Zur Terminierung: Sie dürfen davon ausgehen, dass das Ermitteln eines Zyklus in G in endlich vielen Schritten möglich ist. Zur Korrektheit: Beweisen Sie die Invariante, dass der Restgraph $(V, E\backslash\{e\})$ nach Entfernen der schwersten Kante $e \in C$ noch einen MST vom Ursprungsgraphen (V, E) enthält.