Sven Köhler
Felix Schiller

Freiburg, den 05. Februar 2016
Besprechung 12. Februar 2016

Übungen zur Vorlesung

Graphentheorie

Winter 2015/16

Blatt 7

Aufgabe 1: (4 Punkte)

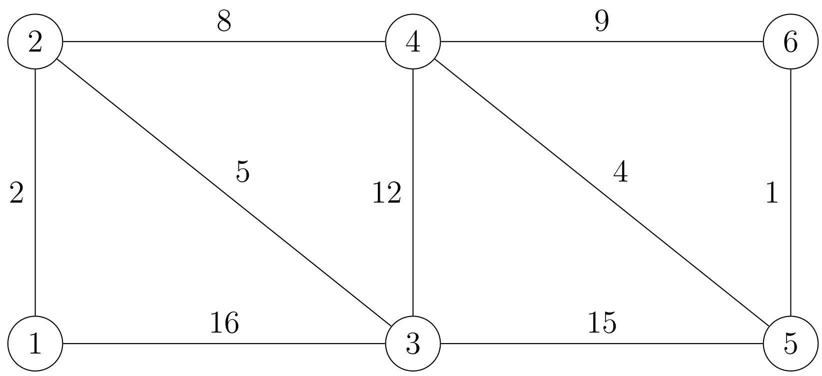

Abbildung 1: Ein einfacher ungerichteter Graph

1. Betrachten Sie den gegebenen Graphen. Wenden Sie den Algorithmus von Kruskal an. Geben Sie die Kanten des berechneten minimalen Spannbaums an und nennen Sie die Reihenfolge, in welcher die Kanten dem Baum hinzugefügt wurden.

2. Wenden Sie den Algorithmus von Prim auf den gegebenen Graphen an und erläutern Sie das unterschiedliche Ergebnis.

Aufgabe 2: (6 Punkte)

Sei G ein zusammenhängender Graph. Ein Partialgraph von G, in dem alle Knoten von einem beliebigen Knoten erreichbar sind, nennen wir *Spanngraph*. Ein Spanngraph heißt *minimal*, wenn er minimale Gesamtkosten hat. Nehmen Sie an, dass negative Kantenkosten erlaubt sind.

1. Beweisen oder widerlegen Sie, dass jeder minimale Spanngraph ein Baum ist.

2. Geben Sie einen Algorithmus im Pseudocode an, welcher minimale Spanngraphen berechnet. Sie können auf den Algorithmen von Prim oder Kruskal aufbauen.

3. Wie müssen die Algorithmen aus der Vorlesung modifiziert werden, um einen maximalen Spannbaum zu berechnen?

Algorithm 1 Algorithmus von Kruskal

Eingabe: $G = (V, E)$, $c : E \to \mathbb{R}$
Ausgabe: Kantenmenge E_F des MSF

```
Sortiere Kanten nach Gewicht: c(e_1) ≤ c(e_2) ≤ ... ≤ c(e_m)
E_F := ∅
for i = 1, 2, ..., m do
    if (V, E_F ∪ e_i) kreisfrei then
        E_F := E_F ∪ {e_i}
    end if
end for
return E_F
```

Algorithm 2 Algorithmus von Prim

Eingabe: $G = (V, E)$ zusammenhängend, $c : E \to \mathbb{R}$
Ausgabe: Kantenmenge E_T des MST

```
Wähle s ∈ V beliebig
E_T := ∅
S := {s}
while S ≠ V do
    Wähle Kante e = [u, v] ∈ δ(S) mit c(e) minimal
    E_T = E_T ∪ {e}
    S = S ∪ {u, v}
end while
return E_T
```