

## Übungsblatt 3: Bidirektionale Suche

Ausgegeben am 27.11.2020. Geschätzter Zeitaufwand: 1 Woche

### Aufgabe 1: Bidirektionale Breitensuche

Eine Variante der Breitensuche, welche in der Praxis oft verwendet wird, ist die *bidirektionale* Breitensuche. Bei der Suche nach einem *st*-Pfad werden hierbei nicht nur ausgehend von *s*, sondern (abwechselnd) auch von *t* aus neue adjazente Knoten exploriert. Dazu wird zusätzlich zum Zustand der Vorwärtssuche („besucht“-Markierung und Queue der gefundenen Knoten) auch der Zustand der Rückwärtssuche verwaltet. Das Wechseln zwischen den beiden Richtungen kann mit verschiedenen Heuristiken (*Wechselstrategien*) zu geeigneten Zeitpunkten passieren.

Wieso funktioniert das und was muss beachtet werden, damit auch tatsächlich kürzeste Wege gefunden werden? Wann darf zwischen den Richtungen gewechselt werden?

### Aufgabe 2: Implementieren

Implementiere und teste die bidirektionale Breitensuche mit einer Wechselstrategie deiner Wahl.

### Aufgabe 3: Laufzeit Messen und Plotten

Lasse die bidirektionale Breitensuche auf verschiedenen großen Graphen für verschiedene *st*-Paare laufen und miss die Laufzeit. Plote die gemessenen Laufzeiten. Kannst du dein Experimentsetup hierfür passend erweitern? Wie schnell ist die bidirektionale Suche im Vergleich zur unidirektionalen Suche auf unterschiedlichen Graphen?