

## Aufgabenblatt 12

letzte Aktualisierung: 17. July, 08:59 Uhr

Ausgabe: 14.07.2017

Abgabe: 31.07.2017 19:59

**Thema:** Bonusblatt: Traveling Salesman Problem

### Abgabe

Die folgenden Datei muss für eine erfolgreiche Abgabe im svn Ordner  
Tutorien/txx/Studierende/deinname@TU-BERLIN.DE/Abgaben/  
eingescheckt sein:

#### Geforderte Datei:

Blatt12/src/TokenGridGraph.java Aufgabe 1.1

### Wichtige Ankündigungen

- Dies ist das letzte Blatt im Semester und ihr könnt mit dieser Aufgabe Bonuspunkte sammeln. Die Aufgabe ist jedoch schwerer als die vorherigen und erfordert, dass ihr die Lösung selbstständig erarbeitet!
- Wenn ihr die erste Klausur am 24.7. schreiben wollt, meldet euch jetzt für den ersten Klausurtermin in OSIRIS an! Wenn ihr nur die zweite Klausur am 6.10. schreiben wollt, meldet euch ab dem 24.7. in OSIRIS an.
- Meldet euch auf keinen Fall für den zweiten Klausurtermin im QISPOS an, falls ihr nur die zweite Klausur schreiben wollt!
- Unit tests dürfen geteilt werden. Lösungen dürfen auf keinen Fall geteilt werden! Wir testen auf Plagiate.

### 1. Aufgabe: Das Traveling Salesman Problem

Das Traveling Salesman Problem (oder auch Problem des Handlungsreisenden) ist ein weiteres Optimierungsproblem auf gerichteten Graphen mit Kantengewichten:

Gesucht ist der Pfad mit minimalen Kosten (Reisezeit), der alle Wegpunkte (Städte) enthält.

**Hinweis:** Das Problem gehört zu der Klasse der so genannten NP-harten Probleme: Es gibt keinen bekannten Algorithmus, der NP-harte Probleme in polynomieller Zeit (vollständig und korrekt) löst. Es gibt aber näherungsweise Lösungsansätze, so genannte Heuristiken.

Auf dieser Aufgabe könnt ihr euch an diesem Problem versuchen. Anders als bei den vorigen Aufgaben gibt es keine in praktikabler Zeit berechenbare optimale Lösung sondern ihr müsst mit Heuristiken arbeiten, um gute Ergebnisse zu erzielen.

#### 1.1. Finden einer Tour im GridGraph (100 Punkte (Bonus))

Die Vorgabe enthält einige rasterförmige (als PNG Bild gespeicherte) Beispielgraphen. Wie schon in der vorherigen Aufgabe stehen weiße Pixel für existierende Knoten. Zusätzlich markieren grüne Pixel ("Token") Knoten, die auf dem resultierenden Pfad (Tour) liegen sollen.

Ändert die Methode `computeTour()` der Klasse `TokenGridGraph` mit Hilfe einer selbstgewählten Heuristik!

Dokumentiert klar in Kommentaren, welche Heuristik ihr angewandt habt und was eure Quelle ist!

**Hinweis:** Um eine Lösung für das Traveling Salesman Problem zu finden gibt es viele verschiedene Ansätze. Es gibt keine Algorithmen mit polynomieller Laufzeit, die eine optimale Lösung generieren, jedoch einige effiziente Heuristiken, die euch eine gute Lösung in polynomieller Laufzeit liefern.

Hier findet ihr eine Liste von Lösungsansätzen, ihr dürft aber auch andere implementieren. Alle diese Methoden könnt ihr mit euren Wissen aus AlgoDat implementieren. Für den Erhalt der Bonuspunkte reicht es aber aus, einen der Ansätze zu implementieren und unsere Vorgabe zu verbessern.

- <https://de.wikipedia.org/wiki/Nearest-Insertion-Heuristik>
- <https://de.wikipedia.org/wiki/MST-Heuristik>  
bzw. [https://de.wikipedia.org/wiki/Algorithmus\\_von\\_Christofides](https://de.wikipedia.org/wiki/Algorithmus_von_Christofides)
- <https://de.wikipedia.org/wiki/K-Opt-Heuristik>  
bzw. [http://akira.ruc.dk/~keld/research/LKH/LKH-1.3/DOC/LKH\\_REPORT.pdf](http://akira.ruc.dk/~keld/research/LKH/LKH-1.3/DOC/LKH_REPORT.pdf)

### 2. Aufgabe: Zusammenfassung Graphenalgorithmen

Besprecht im Tutorium alle behandelten Algorithmen. Wiederholt, wann sie angewendet werden können und wie sich ihre Laufzeit unterscheidet.