

## Modulbeschreibung

# Algorithmen

**Allgemeine Informationen****Anzahl ECTS-Credits**

3

**Modulkürzel**

TSM\_Alq

**Version**

14.04.2017

**Modulverantwortliche/r**

Eric Taillard, HES-SO

**Sprache**

	Lausanne	Bern	Zürich
Unterricht	<input type="checkbox"/> E <input checked="" type="checkbox"/> F	<input type="checkbox"/> D <input type="checkbox"/> E <input type="checkbox"/> F	<input checked="" type="checkbox"/> D <input type="checkbox"/> E
Unterlagen	<input type="checkbox"/> E <input checked="" type="checkbox"/> F	<input type="checkbox"/> D <input type="checkbox"/> E <input type="checkbox"/> F	<input type="checkbox"/> D <input checked="" type="checkbox"/> E
Prüfung	<input type="checkbox"/> E <input checked="" type="checkbox"/> F	<input type="checkbox"/> D <input type="checkbox"/> E <input type="checkbox"/> F	<input type="checkbox"/> D <input checked="" type="checkbox"/> E

**Modulkategorie**

- Erweiterte theoretische Grundlagen
- Technisch-wissenschaftliche Vertiefung
- Kontextmodule

**Lektionen**

- 2 Vorlesungslektionen und 1 Übungslektion pro Woche
- 2 Vorlesungslektionen pro Woche

**Kurzbeschreibung /Absicht und Inhalt des Moduls in einigen Sätzen erklären**

Dieses Modul stellt verschiedene Kategorien von fortschrittlichen Algorithmen und ihre typischen Anwendungsbereiche vor. Im ersten Teil werden die Kenntnisse der dynamischen Datenstrukturen vertieft, die es erlauben, sehr grosse, komplexe oder dynamische Datenmengen oder eine Kombination aus diesen drei Merkmalen effizient zu verwalten. Am Ende des Moduls sind die Studierenden in der Lage, den besten Algorithmus auszuwählen und ihn auf Aufgaben wie die Indexierung, die Suche, die Extraktion, das Einfügen oder die Aktualisierung umfangreicher Datenmengen anzuwenden, wie sie in geografischen Informatiksystemen, in Hypertexten, in der künstlichen Intelligenz oder in der Simulation von Molekülen angetroffen werden. Im zweiten Teil des Moduls werden grundlegende Techniken für den Entwurf von Algorithmen zur Lösung schwieriger Optimierungsprobleme vorgestellt. Die Kombination dieser grundlegenden Techniken – Modellierung, Lösungsbausteine, Lösungsverbesserungen, Problemgliederung – führen zu klassischen Metaheuristiken wie genetische Algorithmen, künstlichen Ameisenkolonien oder Tabusuche. Am Ende des Moduls werden die Studierenden in der Lage sein, diese Techniken zur Lösung schwieriger Optimierungsprobleme zu entwerfen und anzuwenden.

**Ziele, Inhalt und Methoden****Lernziele, zu erwerbende Kompetenzen**

Dieses Modul liefert einen Gesamtüberblick über die verschiedenen, in der Praxis oft angewendeten Klassen von Algorithmen. Auf der Basis solider Kenntnisse sind die Studierenden in der Lage, die effizientesten Algorithmen zu verstehen und einzusetzen und die geeignetsten für ihre eigenen Anwendungen auszuwählen. Die Studierenden:

- kennen die verschiedenen Kategorien von fortgeschrittenen Algorithmen und ihre Anwendungsbereiche;
- verfügen über gute Kenntnisse in fortgeschrittenen Datenstrukturen und deren Verwendung für eine effiziente Verwaltung umfangreicher, komplexer und/oder dynamischer Daten;
- sind in der Lage abschätzen zu können, welche Algorithmen sich für die Lösung bestimmter Aufgaben wie die Indexierung, die Suche, die Extraktion, das Einfügen oder die Aktualisierung umfangreicher Datenmengen eignen;
- kennen die dynamischen Algorithmen, die in der Robotik, in der künstlichen Intelligenz oder in den Molekularwissenschaften benutzt werden.

**Modulinhalt mit Gewichtung der Lehrinhalte**

- Computational Geometry und multidimensionale Datenstrukturen. Gewichtung 50 %.
  - Geometrische Algorithmen
  - Computational Geometry
  - Multidimensionale Datenstrukturen und zugehörige Algorithmen
- Metaheuristische Algorithmen. Gewichtung 50 %.
  - konstruktive Methoden
  - lokale Suche
  - Lerntechniken
  - klassische Metaheuristik: GRASP, künstliche Ameisenkolonien, Tabusuche, Simulated Annealing, Noising-Methoden, genetische Algorithmen, Partikelschwärme usw.

Thema
Teil I: Computational Geometry Anknüpfen an Vorwissen: <ul style="list-style-type: none"> <li>• Asymptotische Notation</li> <li>• Komplexität von rekursiven Algorithmen</li> <li>• Basis-Datenstrukturen (Tabelle, Liste, Stack, Warteschlange, Binärbaum, Heap, Hash-Tabelle)</li> </ul>
Anknüpfen an Vorwissen: <ul style="list-style-type: none"> <li>• Basisalgorithmen (Sortieren, Indexieren und eindimensionale Suche)</li> <li>• Graphen und Netzwerke</li> <li>• Lineare Algebra</li> </ul>
<ul style="list-style-type: none"> <li>• Einführendes Problem: Sichtbarkeit (Sichtbarkeitskarte, Polygone, Boolesche Operationen, Segmentschnittpunkt, numerische Probleme)</li> <li>• Elementare und primitive Objekte (Punkte, Segmente, Polygone, Skalar- und Vektorprodukte, Schnittpunkt, vorhandene Bibliotheken)</li> </ul>
<ul style="list-style-type: none"> <li>• Plane-Sweep-Verfahren (Problem der beiden am nächsten liegenden Punkte, Erkennung von Segmentschnittpunkten)</li> <li>• Konvexe Hülle (ein vollständiger Algorithmus für Punkte in der Ebene; Überblick über andere Paradigmen von Algorithmen und Anwendungen)</li> </ul>
<ul style="list-style-type: none"> <li>• Range-Trees für die orthogonale Suche (Beispiel einer Struktur mit mehreren Ebenen)</li> <li>• Quadrees</li> <li>• Intervallbaum für horizontale Liniensegmente</li> </ul>
<ul style="list-style-type: none"> <li>• Doppelt verkettete Kantenliste (DCEL) und Anwendung auf das Problem der Überlagerung von Karten</li> <li>• Dualität Punkt-Gerade und Anordnung der Geraden</li> </ul>
<ul style="list-style-type: none"> <li>• Triangulation von Polygonen (Überblick über und Algorithmen für monotone Polygone)</li> <li>• Definition von Voronoi-Diagramm, Delaunay-Triangulation und Anwendungen</li> </ul>
Teil II: Metaheuristik Einführung und Anknüpfen an Vorwissen: <ul style="list-style-type: none"> <li>• Grundlegende Probleme und Algorithmen für Graphen und Netze</li> <li>• Optimale Bäume, Wege und Ströme, lineare Zuordnung</li> <li>• Problemmodellierung und kombinatorische Optimierung</li> <li>• Schwierige Probleme: Travelling Salesman, Steiner-Baum, quadratische Zuordnung, Graphenfärbung, Ablaufplanung</li> </ul>
Konstruktive Methoden <ul style="list-style-type: none"> <li>• Wahlloses Bauen</li> <li>• Gieriges Bauen</li> <li>• Bündelsuche, Pilotmethode</li> </ul>
Lokale Suche <ul style="list-style-type: none"> <li>• Nachbarschaftsstruktur</li> <li>• Nachbarschaftsbegrenzung</li> <li>• Nachbarschaftsausweitung, Spektrumssuche, Vertreibungsketten</li> </ul>
Randomisierte Methoden <ul style="list-style-type: none"> <li>• Schwellenwertannahme, Sintflut- und Dämonalgorithmen</li> <li>• Simulated Annealing</li> <li>• Noising-Methoden</li> <li>• GRASP</li> <li>• Variable Nachbarschaftssuche</li> </ul>
Zerlegungsmethoden <ul style="list-style-type: none"> <li>• POPMUSIC</li> </ul> Lernmethoden für Lösungsbausteine <ul style="list-style-type: none"> <li>• Künstliche Ameisenkolonien</li> <li>• Wörterbuchaufbau</li> </ul>
Lernmethoden für die Lösungsverbesserung <ul style="list-style-type: none"> <li>• Tabusuche</li> </ul>

Methoden mit einer Lösungspopulation

- Genetische Algorithmen
- Streuungssuche
- GRASP mit Neuverbindung der Wege
- Schwarmartikel

Das Modul wird in zwei unterschiedliche Themenbereiche (Computational Geometry und Metaheuristik) unterteilt und durch den gleichen Dozenten unterrichtet.

#### Lehr- und Lernmethoden

- Vorlesungen
- Präsentation und Diskussion von Fallstudien
- Übungen zur Theorie und Programmieraufgaben für zuhause

#### Voraussetzungen, Vorkenntnisse, Eingangskompetenzen

Die Studierenden verfügen über Grundkenntnisse in

- Geometrie, linearer Algebra, Algorithmen (Sortieren, Suchen, Hashing) und Datenstrukturen (lineare Strukturen, Baumstrukturen);
- der Graphentheorie (Datenstrukturen und Algorithmen);
- der Komplexität von Algorithmen, Logik, Wahrscheinlichkeitstheorie.

Diese Themen werden im Allgemeinen in Einführungsbüchern zu Algorithmen behandelt. Die Kapitel 1-12, 15-17, 22-26, 28-29, 34-35 von [Cormen 09] vermitteln zum Beispiel die vorausgesetzten Kenntnisse sehr gut.

#### Bibliografie

**M. de Berg, G. Cheong, M. van Kreveld, M. Overmars.** *Computational Geometry : Algorithms and Applications*, Springer, Third Edition 2008.

**T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein.** *Introduction to Algorithms*, third edition, MIT Press, 2009.

**É. D. Taillard,** *Introduction aux métaheuristiques*, 2016.

**P. Siarry (ed.),** *Métaheuristiques*, EAN13 : 9782212139297, EPUB 9782212266214, Eyrolles, 2014

**H. H. Hoos, Th. Stützle,** *Stochastic Local Search: Foundations and Applications*, Morgan Kaufmann / Elsevier, 2004.

## Leistungsbewertung

#### Zulassungsbedingungen für die Modulschlussprüfung (Testatbedingungen)

#### Schriftliche Modulschlussprüfung

Prüfungsdauer : 120 Minuten

Erlaubte Hilfsmittel: Bücher, Folienkopien (ohne Lösungen zu den Übungen)