

Objektorientierte Programmierung von Heuristiken

Object-oriented programming of heuristics

Veranstaltungsnummer | ...

Abschluss des Studiengangs | Master

Semester | Winter

Kurstyp | Vorlesung und Übung

Teilnehmerbeschränkung | Keine

Anrechenbar für | Wahlpflicht (BA&OR)

Unterrichtsstunden | 2 SWS

Leistungspunkte | 5 ECTS

Sprache | Englisch

Verantwortlicher Lehrstuhl | ABWL, SCM und Operations

Referent/en | Dr Andreas Popp

Lernziele/Kompetenzen

Studierende

- wenden die Methoden der objektorientierten Programmierung an.
- können die Vor- und Nachteile der objektorientierten Programmierung gegenüber anderen Programmierparadigmen beurteilen.
- entwickeln strukturierte Programmierprojekte zur Lösung von umfangreichen wissenschaftlichen und praktischen Problemen.
- sind in der Lage die Methoden der objektorientierten Programmierung auf reale Problemstellungen zu übertragen.
- erkennen die im Kurs vorgestellten Algorithmen und können die Probleme schildern, welche dadurch gelöst werden.

Modulinhalt

- Wichtige Programmierparadigmen
- Wiederholung grundlegender Programmierstrukturen
- Grundlagen der objektorientierten Programmierung: Klassen, Felder, Methoden
- Vererbung
- Verwendung externer Klassen
- Generische Klassen
- Alle Inhalte werden am Beispiel wohlbekannter Heuristiken, sowie anderen Algorithmen erläutert

Lehrmethoden

- Vorlesung
- Diskussion
- Interaktive Übung
- Projektarbeit

Prüfungsmodalitäten

- Projektarbeit (100%)

Erläuterung der Prüfungsmodalitäten

- Die Studierenden erhalten einen in der Literatur bekannten Algorithmus und müssen diesen mit den gelernten Methoden implementieren. Abzugeben sind die Implementierung und ihre Dokumentation. Das Projekt muss danach in der Gruppe vorgestellt werden. Alle Elemente werden bewertet.
- Die Implementierung (50% der Endnote) wird bewertet nach Korrektheit, Verständlichkeit und Anwendung von objektorientierten Methoden
- Die Dokumentation und der Vortrag (je 25% der Endnote) werden bewertet nach Vollständigkeit und Verständlichkeit

Arbeitsaufwand (in Stunden)

30 h = Präsenzzeit Vorlesung & Übung

30 h = Vor- und Nachbereitung der Vorlesungsinhalte

20 h = Vor- und Nachbereitung Übungsaufgaben

40 h = Implementierung der Projektarbeit

20 h = Dokumentation der Projektarbeit

10 h = Vorbereitung der Präsentation zur Projektarbeit
150 h = Arbeitsaufwand gesamt

Empfohlene Vorkenntnisse/Voraussetzung laut Prüfungsordnung

- Sicherer Umgang mit dem Computer
- Grundlegende Programmiererfahrung, Kenntnis der folgenden Konzepte: Ausdrücke, Operatoren, Datentypen, Variablen, Bedingte Verzweigung („if-Ausdruck“), Schleifen
- Bereits erlangte Kenntnisse über gängige Heuristiken von Vorteil

Literatur

- Schildt, H.: Java, A Beginner's Guide, 5th Edition, Mcgraw-hill, 2011
- Habelitz, H.: Programmieren lernen mit Java: Aktuell zu Java 7, Galileo Press GmbH, 2012

Object-oriented programming of heuristics

Objektorientierte Programmierung von Heuristiken

Module Number | ...

Degree | Master

Semester | Winter

Course Type | Lecture and tutorial

Participation limit | None

Course Category | Compulsory elective (BA&OR)

Contact Hours | 2 SWS

Number of Credits | 5 ECTS

Language | English

Chair | ABWL, SCM and Operations

Lecturer | Dr Andreas Popp

Learning Outcomes

Students

- apply the methods of object-oriented scheduling.
- can evaluate the advantages and disadvantages of object-oriented programming compared to other programming paradigms.
- develop structured programming projects to solve large-scale scientific and practical problems.
- are able to apply the methods of object-oriented programming to real-world problems.
- recognize the algorithms learned in this course and know the problems solved by them.

Module Content

- Important programming paradigms
- Recapitulation of basic programming structures
- Basics of object-oriented programming: classes, fields and methods
- Inheritance
- Using external classes
- Generic classes
- The content is exemplified with well-known heuristics and other algorithms

Teaching Methods

- Lecture
- Discussion
- Tutorial
- Project assignment

Grading

- Project assignment (100%)

Assessment criteria in detail

- Students get assigned a well-known algorithm from literature must implement it with the learned methods. The implementation and its documentation must be handed in. After that, the project must be presented to the group. All elements are graded.
- The implementation (50% of final grade) will be assessed by correctness, understandability and application of object-oriented methods
- The documentation and presentation (25% of final grade each) are assessed by completeness and understandability

Average Workload

30 h = Time of attendance lecture and tutorial

30 h = Preparation and postprocessing of lecture

20 h = Preparation and postprocessing of exercises

40 h = Implementing the assignment

20 h = Documenting the assignment

10 h = Preparation for the presentation of the assignment

150 h = Total workload

Previous Knowledge/Prerequisites

- Reliable handling of computers
- Basic programming experience, knowledge of the following concepts: expressions, operators, data types, variables, conditional branching („if-clauses“), loops
- Already gained knowledge of common heuristics is an advantage

Readings

- Schildt, H.: Java, A Beginner's Guide, 5th Edition, Mcgraw-hill, 2011
- Habelitz, H.: Programmieren lernen mit Java: Aktuell zu Java 7, Galileo Press GmbH, 2012