

| | |
|---------------------|-------------------------------|
| 03-IMAP-SWRE | Software-Reengineering |
| | <i>Software Reengineering</i> |

Lehrform (*teaching format*) / **SWS** (*hours per week*): 3VL + 1UE

Kreditpunkte (*credit points*): 6

Turnus (*frequency*): i.d.R. jedes WiSe

Inhaltliche Voraussetzungen (*content-related prior knowledge/skills*): Programmierkenntnisse, Erfahrung in der Entwicklung größerer Programme im Team

Sprache (*language*): Deutsch

Lehrende (*teaching staff*): AG Softwaretechnik (Prof. Dr. Rainer Koschke)

| Studiengang (<i>degree program</i>) | Module | Semester |
|--|--------------------------|-----------------|
| Informatik (Master) | IMAP, IMA-SQ, IMVP-VMC | ab 1.Sem. |
| Systems Engineering I/II (Master) | M07-PB-ESS | ab 1./2.Sem. |
| Management Information Systems (Master) | (MIS-INF3) | ab 2.Sem. |
| Informatik (Bachelor) | (nur <i>Freie Wahl</i>) | |

Lernergebnisse:

Die Studierenden verfügen über folgende Fachkompetenzen:

- auf welchen Ebenen man Code analysieren kann,
- wie man Schwachstellen des Codes auffindet,
- wie man duplizierten Code automatisch aufspürt,
- wie man Abhängigkeiten zwischen Anweisungen nachverfolgen kann
- wie man Code-Muster findet,
- wie man den Code automatisch transformieren kann,
- wie man die Stellen im Code findet, die eine bestimmte Funktionalität implementieren,
- wie man Vererbungshierarchien restrukturieren kann,
- wie man Software visualisieren kann,
- wie man Software-Architekturen rekonstruiert
- wie man Reengineering-Projekte organisiert.

Learning Outcome:

Competences in

- how to represent programs
- how to gather information from source code
- how to observe programs at run-time
- how to trace dependencies in programs
- how to detect bad smells including duplicated code
- how to detect patterns in source code
- how to refactor programs and how to transformation programs automatically
- how to reason about programs using formal concept analysis and clustering

- how to locate the code that implements a particular feature
 - how to restructure class hierarchies
 - how to visualize software to enable visual analytics
 - how to reconstruct software architecture from source code
 - how to manage reengineering projects
-

Inhalte:

Software-Reengineering beschäftigt sich mit Wiedergewinnung verlorener Informationen über existierende Software-Systeme (Reverse Engineering), Restrukturierung der Beschreibung des Systems (Refactoring) und der nachfolgenden Implementierung der Änderungen (Alteration). Reengineering hat dabei nicht nur mit alter Software zu tun; gerade neuere objekt-orientierte Systeme erfordern oft schon bald eine Restrukturierung, weshalb sich ein guter Teil der Vorlesung speziell objekt-orientierter Software widmet (Restrukturierung von Klassenhierarchien, automatisches Refactoring).

- allgemeiner Überblick über das Thema sowie Beziehung des Reengineerings zu verwandten Gebieten der Software-Wartung, Wrapping, etc.
- Programmrepräsentationen (Tokenströme, abstrakte Syntaxbäume, Program Dependency Graphs, Static Single Assignment Form)
- statische Programmanalysen (Datenfluss-/Kontrollflussanalysen)
- dynamische Programmanalysen
- Software-Metriken
- Software-Architurrekonstruktion (Reflexionsmethode, Software-Clustering)
- Program Slicing
- Klonerkennung
- Mustersuche
- automatische Code-Transformationen und Refactoring
- formale Begriffsanalyse
- Merkmalsuche
- Analyse und Restrukturierung von Vererbungshierarchien
- Software-Visualisierung
- Planung und Durchführung von Reengineering-Projekten, Prozessmodelle des Reengineerings

Die Übungen dienen, neben der Wiederholung und praktischen Vertiefung des Vorlesungsinhalts, auch der Vorstellung existierender Reengineering-Werkzeuge.

Contents:

- basic terminology in reengineering
 - program representations and static/dynamic program analyses
 - detection of bad smells including duplicated code
 - pattern detection in source code
 - program slicing
 - automated code transformation and refactoring
 - formal concept analysis
 - feature location
 - restructuring of class hierarchies
 - software visualization
 - architecture reconstruction and clustering
 - management of reengineering projects
-

Hinweise (*remarks*): In der Tabelle sind nur die primären/spezifischsten Module aufgelistet, denen diese Veranstaltung zugeordnet ist.