Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Computer Science

- Courses in English* -

- Computer Graphics
- Databases
- Lab for Application Integration (HAWAI)
- Modelling & Simulation Research Project
- Operating Systems
- Seminar (Bachelor)
- Software Construction 2
- Software Engineering

* courses are offered in the summer semester (March – July) only

Collaboration of the Department of Computer Science & Department of Information & Electrical Engineering (Sept. 2016)

Course Name: Introduction to Computer Graphics

Degree programme:
Computer Science (Bachelor programmes)

Responsible Lecturer: Prof. Dr. Philipp Jenke

Work load: 180 hours	**Lecture hours per week:** 2 + 2 hrs lab/week	**ECTS Credits**: 6

Course objectives:

After taking this course the students …

- know how to represent 3-dimensional scenes and how to use the rendering pipeline to create 2-dimensional images from such scenes.
- are able to apply state-of-the-art computer graphics algorithms to solve general modelling, surface representation and rendering problems.
- are able to solve basic computer graphics problems using analytical geometry techniques.
- are able to explain basic data structures to represent surfaces, curves and scenes and choose appropriate candidates for given problems.
- are able to implement solutions related to the above described topics in 3D using OpenGL.

Contents:

- Rendering Pipeline
- Polygonal Representations
- Lighting and Textures
- Implicit Functions
- Curves and Surfaces
- Global Illumination
- Visibility and Data Structures

About didactics and work load distribution:

- Inverted classroom: videos with the lecture content are available for each lecture
- Lecture: Q&A, repetition, exercises, in-depth topics
- Lab exercises deepen the lecture content; working in teams of 2 students; exercises prepared before and presented during lab hours

lecture: 32h, lab: 32h, individual study: 116h

Requirements for participation:

- Java or C# for the lab exercises
- Willingness to get involved in mathematical background (e.g. matrices, vertices, …)
- Ability to imagine 3-dimensional problems and solutions

Type of exam:

Written exam OR practical project (implementation and presentation and elaboration)

Course language:

- English (lecture + slides)
- German (videos)

Requirements for credit point allocation:

- Active participation in lectures and lab
- Passing lab requirements
- Passing written exam OR practical project

Literature:

- Akenine-Moeller, Tomas; Haines, Eric: Real-Time Rendering. 3. edition, A.K. Peters, 2008
- Hughes, John F.; van Dam, Andries; McGuire, Morgan; Sklar, David F.; Foley, James D.; Feiner, Steven K.; Akeley, Kurt: Computer Graphics : Principles and Practice. 2. edition, Amsterdam: Addison-Wesley Longman, 1996
- Virag, Gerhard: Grundlagen der 3D-Programmierung : Mathematik und Praxis mit OpenGL. München : Open Source Press, 2012
- Klawonn, Frank: Grundkurs Computergrafik mit Java : Die Grundlagen verstehen und einfach umsetzen mit Java3D. 3. Auflage, Wiesbaden: Vieweg und Teubner, 2010
- Lecture material: lecture notes, lecture slides

Course Name: Databases

Degree programme:
Information Engineering (Bachelor)

Responsible Lecturer: Prof. Dr. Wilfried Wöhlke

Work load: 180 hours

Lecture hours per week: 3 + 1 hrs lab/week

ECTS Credits: 6

Course objectives:

The students

- have the ability to design a relational database system,
- have the knowledge of Entity Relationship Modeling, Normalization, Structured Query Language.

Contents:

- History
- Database Management Systems
- Entity Relationship Model
- Algebra of Relations
- Normalization
- Structured Query Language

About didactics and work load distribution:

Lecture: Tuition in seminars, blackboard, slides, computer simulation
Laboratory: Laboratory- and computer practical course

attendance: 72h, individual study: 108h

Requirements for participation:

Good knowledge of software construction (This is a 4th semester class)

Course language:

English

Type of exam:

Lecture: Successful passing of written exam

Laboratory: Successful participation of the lab-courses with written reports and short final exam

Requirements for credit point allocation:

- Active participation in lectures and lab
- Passing lab requirements & written exam

Literature:

- Kähler, W.-M. (2008): SQL mit ORACLE, Vieweg Verlag
- • Heuer, A. (2000): Datenbanken Konzepte und Sprachen, mitp Verlag

Course Name: Lab for Application Integration (HAWAI)

Degree programme:	Responsible Lecturers:
Computer Science programmes (Bachelor)	Prof. Dr. Stefan Sarstedt; Prof. Dr. Ulrike Steffens

Work load: 270 hours	**Lecture hours per week:** 6	**ECTS Credits:** 9

Course objectives:

Students will work in small work groups on a development project in the area of business information systems integration into a larger application landscape using typical concepts, methods, and tools for application system integration.

Contents:

Students will independently complete a development project. The project includes:

- Analysis of the integration problem
- Description of existing and planned application landscapes in their business context
- Exploration of existing large of-the-shelf or open-source software applications
- Choice of adequate architecture patterns for application integration
- Implementation of interfaces for application integration
- Presentation of conceptual and technical results in the area of application integration

About didactics and work load distribution:

270 hours of individual study and project work. Students' work is closely related to the HAWAI research project in the Department of Computer Science. The course includes several optional lecture sessions and several team progress review meetings with the lecturers realized in plenary presentation sessions. Final project results will be presented publicly to members of the Department of Computer Science and of the Department of Business.

Requirements for participation:

- Programming experience in an arbitrary programming language is mandatory
- Familiarity with typical software engineering tools is of use

Course language:

English

Type of exam:

Completion and presentation of the project as a team; submission of a complete and meaningful documentation as well as of source code produced throughout the project.

Requirements for credit point allocation:

-

Literature:

- G. Hohpe, B. Woolf: Enterprise Integration Patterns. AddisonWesley, 2003.
- M. Lankhorst: Enterprise Architecture at Work: Modelling, Communication and Analysis. Springer, 2013.

Course Name: Modelling & Simulation Research Project (interdisciplinary)		
Degree programme: **Computer Science** (Bachelor/Master/PhD students)	Responsible Lecturer: Prof. Dr. Thomas Thiel-Clemen	
Work load: 180 hours*	**Lecture hours per week:** N/A	**ECTS Credits:** 6*

Course objectives:

Students will be part of the MARS research group (http://mars-group.mars.haw-hamburg.de/en/) where they will work on an individual research and development project suitable to their level. A major objective of this module is to learn how to complete collaborative research in a larger team.

Note: The classes in this module handbook are undergraduate classes. It is however possible for Master or PhD students to do a semester-long research project within the MARS group only.

Contents:

Self-learning materials and coaching will be provided to students during the semester. A highly experienced team is also available. The semester covers:

- an introduction to modeling & simulation
- how to sharpen the research question
- conceptual modeling
- how to select the right simulation framework
- verification & validation

About didactics and work load distribution:

Individual studies and collaborative project work. A research and milestone plan will be created at the beginning of the module.

* The workload of this project can be increased to 12 credits, so that together with other modules it can make up a total semester workload of 30 ECTS.

Requirements for participation:	**Course language:**
Successful completion of elementary studies. Interest in interdisciplinary research issues.	English
Type of exam: Conference or journal paper draft, ready for submission	

Requirements for credit point allocation:

-

Literature:

-

Course Name: Operating Systems

Degree programme: **Information Engineering** (Bachelor)	**Responsible Lecturer:** Prof. Dr.-Ing. Holger Gräßner	
Work load: 180 hours	**Lecture hours per week:** 3 + 1 hrs lab/week	**ECTS Credits**: 6

Course objectives:

The students have

- an overview about existing operating systems and their individual characteristics,
- the ability to use different OS resources in order to program dedicated application tasks,
- the ability to design and realize complex systems using the available OS resources.

Contents:

- Multitasking methods
- Communication and synchronization
- Resource sharing and timing control
- Interaction with external signals
- I/O programming, OS driver basics
- OS comparison and selection
- Selected topics in modern OS
- Exemplary applications during the lab with in-depth system-analysis and realisation

About didactics and work load distribution:

- Lecture: Q&A, repetition, exercises, in-depth topics
- Lab exercises deepen the lecture content; working in teams of 2 students; exercises prepared before and presented during lab hours

attendance: 72h, individual study: 108h

Requirements for participation:

Basic knowledge of software construction, computer architecture (This is a 6th semester class)

Course language:

English

Type of exam:

Lecture: Successful passing in written exam
Laboratory: Successful participation in lab-courses with lab preparations with reviews, functional programmes, lab reports

Requirements for credit point allocation:

- Active participation in lectures and lab
- Passing lab requirements
- Passing written exam OR practical project

Literature:

- Tanenbaum, A.S. (2009): Modern Operating Systems, Prentice Hall
- Kernighan, B.W.; Ritchie, D. M. (2000): The C-Programming Language (ANSI C), Markt+Technik Verlag
- Kerrisk, M. (2010): The Linux Programming Interface, No Starch Press
- Corbet, J. et al. (2005): Linux Device Drivers, O'Reilly

Course Name: Seminar (Bachelor)

Degree programme: **Computer Science** (Bachelor)	Responsible Lecturer: Prof. Dr. Zhen Ru Dai

Work load: 90 hours	**Lecture hours per week:** 2	**ECTS Credits:** 3

Course objectives:

On completion of the seminar the student will have

- familiarization with a new computer science topic
- become acquainted with a given technical and scientific topic
- prepared presentation slides for the topic
- given an understandable presentation to an audience that is not familiar with the special computer science topic
- learned techniques about presentation, discussion and evaluation
- learned to work with presentation tools e.g. Powerpoint

Contents:

This module is an excellent starting point for student research either individually or as part of a research team. Students will learn how to dive deeply into a scientific topic, present the key ideas in front of a peer group, and react to feedback.

- Presentation techniques
- Investigate a technical topic
- Tool handling
- Discussion and evaluation
- Provide feedback to presenter
- work out outline in languages English and/or German

About didactics and work load distribution:

- presentation of 30 minutes, discussion and feedback from the audience
- support in working out the presentation slides
- test presentation with supervisor
- write summary as outline

Requirements for participation:

4 semesters of Computer Science and higher

Type of exam:

Delivery of presentation slides and abstracts

Course language:

German and English
(presentations by students depending on nationality)

Requirements for credit point allocation:

compulsory attendance

Literature:

- Martin Hartmann, Rüdiger Funk, Horst Nietmann: Präsentieren; Beltz
- Josef W. Seifert: Visualisieren, Präsentieren, Moderieren; Gabal
- Christian W. Dawson: Computerprojekte im Klartext; Pearson Studium
- Christian W. Dawson: Projects in Computing and Information Systems, Pearson

Course Name: Software Construction 2

Degree programme: **Information Engineering** (Bachelor)	**Responsible Lecturer:** Prof. Dr. Marc Hensel

Work load: 180 hours	**Lecture hours per week:** 3 + 1 hrs lab/week	**ECTS Credits**: 6

Course objectives:

The students
- understand JAVA syntax and can write a JAVA program,
- can construct classes in object oriented form using the JAVA API,
- are able to design and test JAVA programs inside a development tool,
- are able to use encapsulation and inheritage structures,
- can use packages, streams, file handling, threads, swing and other parts of the basic JAVA API,
- can construct JAVA software including a graphical user interface for small applications.

Contents:

Lecture:
- Introduction into the object oriented programming in JAVA
- The Programming environment and the fundamental programming structures in JAVA
- The object oriented programming fundamentals
- The basic usage of classes, associations, inheritance, encapsulation and other object oriented subjects
- Main libraries of the API (Application Programming Interface)
- The execution of JAVA programs using graphical user interfaces and threads

Lab:

During the laboratories students learn how to the transfer the main parts of the object-oriented JAVA syntax into applications. The implementation of JAVA programs, the usage of JAVA classes and the usage of the JAVA software Developers Kit (SDK) is the main focus of this module.

About didactics and work load distribution:
- Lecture: Q&A, repetition, exercises, in-depth topics
- Lab exercises deepen the lecture content; working in teams of 2 students; exercises prepared before and presented during lab hours

attendance: 72h, individual study: 108h

Requirements for participation:

Basic knowledge of software construction (This is a 2nd semester class)

Course language:

English

Type of exam:

Lecture: Successful passing in written exam

Laboratory: Successful participation in the lab-courses with written reports and a final exam

Requirements for credit point allocation:
- Active participation in lectures and lab
- Passing lab requirements
- Passing written exam OR practical project

Literature:

- Haines, S.; Potts, S. (2002): Java 2 Primer Plus, SAMS Publishing
- Flanagan, D. (2005): JAVA in a Nutshell, A Desktop Quick Reference, O'Reilly
- Horstmann, C.S.; Cornell, G. (2003): Core Java 2, Volume I-Fundamentals, Sun Microsystems Press
- Esser, F. (2001): Java 2, Designmuster und Zertifizierungswissen, Galileo Press
- Eckel, B. (2006): Thinking in Java, Prentice Hall
- Arnold, K.; Gosling, J.; Holmes, D. (2001): The Java Programming Language Third Edition, Addison-Wesley

Course Name: Software Engineering

Degree programme: **Information Engineering** (Bachelor)	**Responsible Lecturer:** Prof. Dr. Marc Hensel

Work load: 180 hours	**Lecture hours per week:** 3 + 1 hrs lab/week	**ECTS Credits**: 6

Course objectives:

The students
- have the ability to analyze applications and to realize a requirement analysis,
- have the ability to describe applications within the UML (Unified Modeling Language),
- can identify the relationship and associations inside applications,
- can use case studies to design class and sequence diagrams,
- can transfer application description into an object oriented program description,
- can design the software for small applications using different software engineering models especially different prototyping models.

Contents:

- This unit introduces into the basic ideas of the software engineering process and the UML (Unified Modelling Language)
- The goal is to construct object oriented software for applications using software engineering methods.
- Especially the module focuses on the requirement analysis, the use case study, the sequence and collaboration diagram construction and several other software engineering development diagrams inside a software engineering tool based on UML
- The whole process starting with the analysis, via the requirements, the design and realization of software applications is described
- Different software development and design models are lined out
- All the theoretically knowledge earned has to be transferred into the software construction process for small applications

About didactics and work load distribution:

Lecture: Tuition in seminars, blackboard, slides, computer simulation
Laboratory: Laboratory- and computer practical course

attendance: 72h, individual study: 108h

Requirements for participation:

Very good knowledge of software construction, object-oriented software construction in JAVA (This is a 4th semester class)

Course language:

English

Type of exam:

Lecture: Successful passing of written exam

Laboratory: Successful participation of the lab-courses with written reports and short final exam

Requirements for credit point allocation:

- Active participation in lectures and lab
- Passing lab requirements & written exam

Literature:

- Burckhardt, R. (1999): UML Unified Modeling Language, objektorientierte Modellierung für die Praxis, Addison-Wesley
- Booch, G.; Rumbaugh, J.; Jacobson, I. (1999): The Unified Modeling Language User Guide, Addison-Wesley
- Douglass, B.P. (2004): Real Time UML: Advances in the UML for Real-Time Systems, Addison-Wesley
- Oestereich, B. (2006): Objektorientierte Softwareentwicklung, Analyse und Design mit UML 2.1, Oldenbourg Verlag
- Rumbaugh, J.; Jacobson, I.; Booch, G. (2010): The Unified Modeling Language Reference Manual, Addison-Wesley
- Sommerville, I. (2010): Software Engineering, Addison-Wesley