

Course

AD - algorithms and data structures

Version: 2 | Last Change: 10.09.2019 15:53 | Draft: 0 | Status: vom verantwortlichen Dozent freigegeben

^ General information

Long name	algorithms and data structures
Approving CModule	AD_BaTIN
Responsible	Prof. Dr. Dieter Rosenthal Professor Fakultät IME
Level	Bachelor
Semester in the year	summer semester
Duration	Semester
Hours in self-study	78
ECTS	5
Professors	Prof. Dr. Dieter Rosenthal Professor Fakultät IME
Requirements	basics of programming in high level programming languages
Language	German
Separate final exam	Yes

Final exam

Details

Students shall prove that they can 1.) explain and apply fundamental terms, concepts, and techniques, 2.) apply programming and more abstract concepts to solve application problems in the field of data structures and algorithms and 3.) assess the correctness of statements and program code. Typical types of assignments are 1.) multiple choice questions, fill-in-the-blank texts, assessment of statements, 2.) write program code or develop a solution in a more abstract form to solve given problems of limited size and 3.) finding errors in texts and program code.

Minimum standard

At least 50% of the total number of points.

Exam Type

Students shall prove that they can 1.) explain and apply fundamental terms, concepts, and techniques, 2.) apply programming and more abstract concepts to solve application problems in the field of data structures and algorithms and 3.) assess the correctness of statements and program code. Typical types of assignments are 1.) multiple choice questions, fill-in-the-blank texts, assessment of statements, 2.) write program code or develop a solution in a more abstract form to solve given problems of limited size and 3.) finding errors in texts and program code.

^ Lecture / Exercises

Learning goals

Knowledge

concepts of object oriented programming

basics of linear and hierarchic data structures used in programs

linear data structures (e.g. linear lists chained lists)

hierarchic data structures, trees

analysis of the complexity of algorithms

Important search algorithms

mode of operation of important sort algorithms

Skills

apply the concepts of object oriented programming in C++

estimate the quality and useness of data structures an algorithms

implement search and sort algorithms in programs

Expenditure classroom teaching

Type	Attendance (h/Wk.)
Lecture	1
Exercises (whole course)	1
Exercises (shared course)	1

Separate exam

none

^ Practical training

Learning goals

Skills

develop single handed data structures and algorithms

implement linear and hierarchic data structures in C++

implement sort algorithms in C++

implement search algorithms in C++

application of the aspects listed above to real-world scenarios in small teams

Expenditure classroom teaching

Type	Attendance (h/Wk.)
Practical training	1
Tutorial (voluntary)	0

Separate exam

Exam Type

working on practical scenarion (e.g. in a lab)

Details

Students work in small teams. Each team completes multiple "rounds" with assigned appointments in the lab. In each round, programming assignments are solved.

For the preparation of a laboratory appointment a "preparation sheet" has to be solved. The acquired knowledge will be tested at the beginning of the appointment (short written entrance test, interview with the supervisor). In case of failure, a follow-up appointment must be taken; in case of multiple failures, the student will be excluded from the lab. In case of success, a "laboratory work sheet" with further tasks will be worked on under supervision (and, if necessary, with assistance).

Minimum standard

Successful participation in all laboratory appointments, i.e. in particular independent solution (or with some assistance if necessary) of the programming assignments.