

Course Manual INF2

Computer Science 2

Version: 5 | Last Change: 29.09.2019 16:57 | Draft: 0 | Status: vom verantwortlichen Dozent freigegeben

– General information

Long name	Computer Science 2
Approving CModule	INF2 BaMT
Responsible	Prof. Dr.-Ing. Arnulph Fuhrmann Professor Fakultät IME
Valid from	summer semester 2021
Level	Bachelor
Semester in the year	summer semester
Duration	Semester
Hours in self-study	90
ECTS	6
Professors	Prof. Dr.-Ing. Arnulph Fuhrmann Professor Fakultät IME
Requirements	Computer Science 1
Language	German
Separate final exam	Yes

Literature

G. Saake, K. Sattler: Algorithmen und Datenstrukturen, dpunkt.verlag, 2010

R. Sedgewick, K. Wayne: Introduction to Programming in Java, Addison Wesley, 2007

W. Küchlin, A. Weber: Einführung in die Informatik - Objektorientiert mit Java, Springer, 2005

P. Gumm, M. Sommer: Einführung in die Informatik, Oldenbourg, 2010

Final exam

Details

Students must demonstrate the following competences in a written examination:

- Development of short programmes to solve problems described in colloquial language (K.4, K.9)
- Development of short programs to solve abstractly described problems (K.2)
- Application of programming language and more abstract constructs to solve application problems (K.8)
- Reading, understanding and, if necessary, correction of given program fragments (K.4, K.10)
- Evaluation of statements with regard to their correctness (K.11)

Minimum standard

At least 50% of the total number of points.

Exam Type

EN Klausur

– Lecture / Exercises

Learning goals

Goal type	Description
Knowledge	Advanced methods of object orientation polymorphism Abstract Classes interfaces modelling Generic Programming Dynamic data structures concatenated lists stacks cues hash tables trees algorithms intricacy O notation expenditure of time storage effort performance measurement General strategies for designing algorithms brute force greedy divide-and-conquer backtracking sorting methods Selection Sort Insertion Sort Merge Sort search procedure Linear search Binary Search
Skills	Creating object-oriented programs in Java Designing object-oriented models for a given problem Using class diagrams Convert to software dynamic data structures Using dynamic data structures in Java Designing dynamic data structures Implement dynamic data structures in Java Determining the complexity of algorithms Solving a problem using suitable algorithms Selecting algorithms Designing algorithms Implementing Algorithms in Java

Special requirements

none

Accompanying material	electronic lecture slides for the lecture, electronic exercise collection
------------------------------	--

Separate exam	Yes
----------------------	-----

Separate exam

Exam Type	EN Übungsaufgabe mit fachlich / methodisch eingeschränktem Fokus lösen
------------------	--

Details	Independent solving of self-learning tasks on the topics of the lecture in the form of the development of more complex programs to solve problems described in colloquial or abstract language (K.4, K.5, K.9, K.2).
----------------	--

Minimum standard	More than 80% of all exercises submitted. A task is deemed to have been completed if it has been solved predominantly and independently.
-------------------------	--

Expenditure classroom teaching

Type	Attendance (h/Wk.)
Lecture	3
Exercises (whole course)	0
Exercises (shared course)	2
Tutorial (voluntary)	2