

# Lehrveranstaltung

## INF1 - Informatik 1

---

Version: 1 | Letzte Änderung: 27.09.2019 20:30 | Entwurf: 0 | Status: vom verantwortlichen Dozent freigegeben

### ^ Allgemeine Informationen

<b>Langname</b>	Informatik 1
<b>Anerkennende LModule</b>	<u>INF1_BaMT</u>
<b>Verantwortlich</b>	Prof. Dr.-Ing. Arnulph Fuhrmann Professor Fakultät IME
<b>Niveau</b>	Bachelor
<b>Semester im Jahr</b>	Wintersemester
<b>Dauer</b>	Semester
<b>Stunden im Selbststudium</b>	90
<b>ECTS</b>	6
<b>Dozenten</b>	Prof. Dr.-Ing. Arnulph Fuhrmann Professor Fakultät IME
<b>Voraussetzungen</b>	keine
<b>Unterrichtssprache</b>	deutsch
<b>separate Abschlussprüfung</b>	Ja

## Abschlussprüfung

### Details

Die Studierenden weisen in einer schriftlichen Klausur folgende Kompetenzen nach:

- Entwicklung kurzer Programme zur Lösung von abgegrenzten umgangssprachlich beschriebenen Problemen (K.4)
- Entwicklung kurzer Programme zur Lösung von abstrakt beschriebenen Problemen (K.2, K.5)
- Lesen, verstehen und ggf. Korrektur von vorgegebenen Programmfragmenten (K.4, K.10)
- Bewertung von Aussagen hinsichtlich ihrer Korrektheit (K.12)

### Mindeststandard

Mindestens 50% der möglichen Gesamtpunktzahl.

## Prüfungstyp

Die Studierenden weisen in einer schriftlichen Klausur folgende Kompetenzen nach:

- Entwicklung kurzer Programme zur Lösung von abgegrenzten umgangssprachlich beschriebenen Problemen (K.4)
- Entwicklung kurzer Programme zur Lösung von abstrakt beschriebenen Problemen (K.2, K.5)
- Lesen, verstehen und ggf. Korrektur von vorgegebenen Programmfragmenten (K.4, K.10)
- Bewertung von Aussagen hinsichtlich ihrer Korrektheit (K.12)

## ^ Vorlesung / Übungen

### Lernziele

---

#### Kenntnisse

Grundlagen

Computerarchitektur  
von Neumann Modell

Prozessor

Speicher

I/O

Binärcodierung von Daten

Ganze Zahlen

Zeichen und Text

Gleitkommazahlen

Entstehung, verbreiten und verstärken von Fehlern durch Rundungseffekte

Mediendaten

Bilder

Sound

Ausblick auf diese Daten, mehr Details, wenn mehr Grundlagen besprochen

---

Kompiler, Interpreter, Hybride Sprachen

---

Imperative Programmierung

Syntax, Schlüsselwörter, Kommentare

Variablen

Primitive Datentypen

Operatoren und Ausdrücke

Arithmetische Operatoren

Operatoren auf Wahrheitswerten

Operatoren auf Bitmustern

Ausdrücke

arithmetisch

boolsch

Präzedenz von Operatoren

Grundlegende Datenstrukturen

Arrays

Zeichen und Zeichenketten

Referenzen

Kontrollstrukturen  
Flussdiagramme  
Ein / Ausgabe

---

Prozedurale Programmierung  
Strukturierung  
Funktionen  
Rekursion  
Module und Bibliotheken  
Modellierung

---

Objektorientierte Programmierung  
Klassen  
Objekte  
Methoden  
Kapselung  
Vererbung  
Polymorphismus

---

Softwarequalität  
Fehlerbehandlung, Fehlerkorrektur  
Testen  
Dokumentation

---

## Fertigkeiten

Entwurf und Modellierung  
Abstrahieren von Problembeschreibungen in Algorithmen  
Entscheiden, welche Programmierkonstrukte zur Lösung einer Problemstellung geeignet sind  
Entwurf und Modellierung von Softwaresystemen mit UML

---

Programmieren in Java  
Überprüfen von Programmen auf Fehler  
Entwickeln von Programmen zur Lösung von konkreten Problemstellungen  
Anwenden von Programmierkonzepten  
Erfassen der Funktionsweise von Quelltexten Dritter

## Aufwand Präsenzlehre

Typ	Präsenzzeit (h/Wo.)
Vorlesung	3
Übungen (ganzer Kurs)	0
Übungen (geteilter Kurs)	2
Tutorium (freiwillig)	2

# Separate Prüfung

## Prüfungstyp

Übungsaufgabe mit fachlich / methodisch eingeschränktem Fokus lösen

## Details

Selbständiges Lösen von Selbstlernaufgaben zu den Themen der Vorlesung in Form der Entwicklung komplexerer Programme zur Lösung von umgangssprachlich oder abstrakt beschriebenen Problemen (K.4, K.5, K.9, K.2).

## Mindeststandard

Mehr als 80% aller Übungsaufgaben abgegeben. Eine Aufgabe, gilt als abgegeben, wenn diese zum überwiegenden Teil und selbstständig gelöst wurde.