

# Grundzüge der Programmierung - Prüfungsstoff

Stephan Adelsberger, Sebastian Neumaier, Svitlana Vakulenko,  
Anton Yeshchenko

- **Programmieren:** Definition Algorithmus, Syntax, Semantik, Programmiersprachen, Compiler, Interpreter
- **Basiselemente und Typen:** Variablen, Operatoren, primitive Datentypen, Strings, Statische vs. dynamische Typisierung, Explizite / Implizite Typumwandlung
- **Kontrollstrukturen und Funktionen:** Verzweigungen, Schleifen, Funktionen (Parameter, Rückgabotyp)
- **Listen:** Arrays (Listen), Erzeugen und arbeiten mit Arrays, Mehrdimensionale Arrays
- **Klassen:** Klasse vs. Instanz, Objekte, Erzeugen von Objekten, Instanzvariablen, Konstruktoren, Überladen von Methoden
- **Vererbung:** Klassenhierarchie, Überschreiben von Methoden, Problem Mehrfachvererbung und Lösungsansätze
- **Sichtbarkeiten:** Sichtbarkeit von Variablen (Lebensdauer), Sichtbarkeitsattribute (Java) bzw. Namens- und Zugriffskonvention (Python), Getter- und Setter-Methoden
- **Objektorientierte Programmierung:** Zweck OOP, Vererbung ("ist-ein"- und "ist-Teil-von"-Beziehung), Information Hiding/Datenkapselung, abstrakte Klassen, Super-Konstruktor, Polymorphismus

Für Studenten, welche die Prüfung im Rahmen des Wahlfachs ablegen, sind zusätzlich die folgenden Themen bzw. Kapitel prüfungsrelevant.

## Wahlfach:

- **Exceptions**
- **Dynamische Datenstrukturen:** Stacks, Queues, Maps (Dictionaries), Trees
- Umsetzung von Beziehungen zwischen Klassen

# Principles of programming - Exam topics

Stephan Adelsberger, Sebastian Neumaier, Svitlana Vakulenko,  
Anton Yeshchenko

- **Programming:** algorithms, syntax and semantics, programming, compiler, interpreter
- **Basics and types:** variables, operations, primitive data types, Strings, static vs dynamic typing, explicit vs implicit type casting
- **Control flow and functions:** if-else branches, loops, functions (parameters, return values)
- **Lists:** Arrays (lists), create and fill Arrays, multidimensional Arrays
- **Classes:** Class vs Instance of class, objects, create objects, instance variable, constructor, method overloading
- **Inheritance:** inherit classes, method overriding, problems and solutions for multiple inheritance
- **Information hiding:** variable access, access modifier (Java) and naming conventions, get- and set-methods
- **Object oriented programming:** why OOP, inheritance (“is-a”- and “is-part-of”-relations), information hiding/encapsulation, abstract classes, Super-constructor, polymorphism

Additional topics if you plan to take the exam as “Wahlfach” course (5 ECTS):

## Wahlfach:

- **Exceptions**
- **Dynamic data structures:** Stacks, Queues, Maps (Dictionaries), Trees
- How to model (UML) relations in programming languages