

## Allgemeines

<i>Dozent:</i>	Prof. Dr. Theodor Tempelmeier
<i>Verantwortlich:</i>	Prof. Dr. Theodor Tempelmeier
<i>Studiengang:</i>	Master
<i>Pflicht/FWPF:</i>	FWPF
<i>Voraussetzungen:</i>	Grundstudium
<i>Sprache:</i>	Deutsch
<i>Lehrform:</i>	4 SWS seminaristischer Unterricht mit Übungen
<i>Arbeitsaufwand:</i>	60 Stunden Präsenzzeit, 90 Stunden Selbststudium
<i>Leistungspunkte:</i>	5
<i>Medienform:</i>	Vortrag mit Overhead-Projektor, Lückenskript (PDF), Kurzbeschreibung im WWW
<i>Prüfung:</i>	Klausur, 90 Minuten, Unterlagen lt. Aushang

## Lernziele und Inhalt

### Richtziel

Richtziele sind Kenntnisse der Elemente von Programmiersprachen, Verständnis für Hintergründe und Querbezüge im Bereich der Programmiersprachen sowie die Fähigkeit zum Vergleich und zur Einordnung von Programmiersprachen.

### Inhaltsübersicht

Es wird ein Überblick über Programmiersprachen und eine Kurzübersicht über wichtige gängige Programmiersprachen gegeben. Anschließend erfolgt ein ausführliches Studium aller wichtigen programmiersprachlichen Elemente anhand einer Beispielsprache, insbesondere Typkonzepte, Überladen von Operatoren, Modularisierung, generische Programmierung, hardwarenahe Programmierung. Dabei werden Vergleiche mit verschiedenen anderen Programmiersprachen angestellt. Der Schwerpunkt der Veranstaltung liegt auf dem imperativen Programmierstil, doch werden auch Querbezüge zum objektorientierten und funktionalen Programmierstil hergestellt.

### Inhalt (beispielhaft für die Variante mit Ada als Lehrsprache)

1. *Einleitung*
2. *Kurzübersicht über einige praxisrelevante Sprachen*  
Cobol, Fortran 77, Pascal-Sprachfamilie
3. *Ada - Grundkurs*  
Allgemeines, Deklarationen, Schleifen und Sprünge, Bedingte Anweisungen, Unterprogramme, Pakete, Zeiger, Strings, Generischer Mechanismus
4. *Typen*  
Übersicht, Typäquivalenz bzw. Typkompatibilität, Typen in Ada, Polymorphie, Typerweiterung („Vererbung“)
5. *Ada - weitergehende Konzepte*  
Ausnahmen („Exceptions“), Tasks
6. *Diverse Ergänzungen*  
Kurzschlußoperationen in logischen Ausdrücken, Zahlen – Adressen – Bits, Verbunde mit Diskriminanten, Mengen ("Sets"), Binden bei überladenen Symbolen, Zwischencode bei

der Compilierung ("P-Code", "P-Maschine" für Pascal-Programme), Slicing am Beispiel OLAP, Das Problem der Wortlänge, Variante Verbunde ( $\approx$  Unions)

## Literatur

### Besonders empfohlen

1. Watt, D.A.: *Programming Language Concepts and Paradigms*. Prentice Hall (1990). Auch auf deutsch mit einigen Ergänzungen: Hanser (1996)
2. Sebesta: *Programming Languages*. Addison-Wesley, 4th ed.
3. Henning, P.A., Vogelsang, H. (Hrsg.): *Handbuch Programmiersprachen*. Hanser Verlag, München, 2007.
4. *Sprachdefinitionen/Rationales aller Sprachen* (für Ada im Internet)

### Zusätzlich empfohlen

5. Ben-Ari, M.: *Understanding Programming Languages*. John Wiley & Sons, 1996. Out of print. Electronic edition available. <http://stwww.weizmann.ac.il/g-cs/benari/books/index.html>.
6. Sethi, R.: *Programming Languages - Concepts and Constructs*. Addison-Wesley, 2nd ed. (1996)
7. Ghezzi, C. und Jazayeri, M.: *Konzepte der Programmiersprachen*. Oldenbourg (1989)
8. Fischer, A.E., Grodzinsky, F.S.: *The Anatomy of Programming Languages*. Prentice-Hall (1993)
9. Meyer, B.: *Object-Oriented Software Construction*. 2nd ed. New York (1997)
10. Rogat, A.: *Objektorientiertes Programmieren mit C++ und JAVA*. Vorlesung an der BUGH Wuppertal im WS 1997/98.  
<http://www.math.uni-wuppertal.de/~axel/skripte/oop/start.html>  
oder [http://www.forwiss.uni-passau.de/extern/dokumentation/oop/oop\\_fr.html](http://www.forwiss.uni-passau.de/extern/dokumentation/oop/oop_fr.html)
11. Nagl, M.: *Softwaretechnik mit Ada 95. Entwicklung großer Systeme*. Vieweg (1999)
12. Kratzer, K. P.: *Ada. Eine Einführung für Programmierer*. Hanser (1996)
13. Ben-Ari: *Ada for Software Engineers*. J. Wiley & Sons (1998)
14. *Ada-Syntax*: <http://www.adahome.com/Resources/References.html>,  
<http://cuiwww.unige.ch/db-research/Enseignement/analyseinfo/Ada95/BNFindex.html>
15. *Ada-Tutorials und sonstige Ressourcen*. <http://www.adahome.com/>
16. Schader, M. und Kuhlins, S.: *Programmieren in C++*. 5. Auflage. Springer (1998)
17. Sakkinen, M.: *The Darker Side of C++ Revisited*.  
Structured Programming 13, p155-177. (1992)
18. Urban, H. und Jamin, K. W.: *COBOL*. Rowohlt rororo 8157 (1990)
19. Inglis, J.: *COBOL85 for Programmers*. Wiley and Sons (1989)
20. Schmitt, G.: *FORTRAN-Kurs*. Oldenbourg (1990)
21. Rabenstein, D.: *Fortran 90*. Hanser Hanser (1995)
22. Schader, M. und Schmidt-Thieme, L.: *Java. Eine Einführung*. 2. Auflage. Springer (1999)
23. Wilson, I.R. und Addyman, A.M.: *Pascal – Leichtverständliche Einführung*. Hanser (1984)