



**Universität Stuttgart**

**Modulhandbuch**  
**Studiengang Lehramt an Gymnasien (GymPO I) Informatik**  
Prüfungsordnung: 2010

Universität Stuttgart  
Keplerstr. 7  
70174 Stuttgart

## Inhaltsverzeichnis

<b>Präambel</b> .....	<b>3</b>
<b>200 Pflichtmodule</b> .....	<b>4</b>
10280 Programmierung und Software-Entwicklung .....	5
10940 Theoretische Grundlagen der Informatik .....	7

## Präambel

nicht verfügbar

---

## 200 Pflichtmodule

---

Zugeordnete Module:   10280 Programmierung und Software-Entwicklung  
                          10940 Theoretische Grundlagen der Informatik

---

## Modul: 10280 Programmierung und Software-Entwicklung

2. Modulkürzel:	051520005	5. Moduldauer:	1 Semester
3. Leistungspunkte:	9.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	6.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Bernhard Mitschang		
9. Dozenten:	Bernhard Mitschang		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 1. Semester → Basismodule BA (Komb) Informatik, 1. Semester → Module im Nebenfach		
11. Voraussetzungen:	Keine. Teilnahme an einem Mathematik Vorkurs wird empfohlen.		
12. Lernziele:	Die Teilnehmer haben die wichtigsten Konzepte einer höheren Programmiersprache und ihrer Verwendung verstanden und sind in der Lage, kleine Programme (bis zu einigen hundert Zeilen) zu analysieren und selbst zu konzipieren und zu implementieren. Sie kennen die Möglichkeiten, Daten- und Ablaufstrukturen zu entwerfen, zu beschreiben und zu codieren. Sie haben die Abstraktionskonzepte moderner Programmiersprachen verstanden. Sie kennen die Techniken und Notationen zur Definition kontextfreier Programmiersprachen und können damit arbeiten.		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Algorithmen und funktionale Programmierung Kap. 1 verwendet nur die funktionale Teilmenge der Programmiersprache Ada, keine Variablen, keine Prozeduren. Grammatik, Formale Sprachen und BNF werden eingeführt.</li> <li>• Imperative Programmierung Kap. 2 erweitert die verwendete Sprache durch die prozeduralen Konzepte, also Variablen und Prozeduren. Zu den Sprachkonstrukten werden Vor- und Nachbedingungen, mit den Schleifen die Invarianten eingeführt. Datentypen werden schrittweise ausgebaut. In Zusammenhang mit den Zeigern werden die Konzepte für Keller und Halde vermittelt. Die Entwicklung einfacher Programme wird gezeigt und geübt.</li> <li>• Aufbau und Organisation komplexer Programme. Die Modularisierung, die bei größeren Programmen notwendig ist, führt zur Kapselung und zu den abstrakten Datentypen. Damit entsteht die Möglichkeit, neue Datenstrukturen und Datentypen sicher zu definieren. Die Konzepte der Kompilation und der Interpretation werden erläutert. Wichtige Beispiele komplexer Datentypen werden entwickelt. Die Konzepte der Generalisierung (generische Einheiten) werden vermittelt.</li> <li>• Ausnahmebehandlung Möglichkeiten und Probleme der Ausnahmebehandlung sind Gegenstände dieses kurzen Kapitels.</li> <li>• Objektorientierte Programmierung Am Ende des Semesters steht ein Ausblick in die objektorientierte Programmierung, d.h. die Umsetzung der bereits bekannten Konzepte (ADTs) in die objektorientierte Sichtweise und die Vererbung. Dieser Teil bereitet die Programmierung in einer objektorientierten Sprache (3. Semester) vor.</li> </ul>		
14. Literatur:	Manuskripte: V.Claus (WS 08/09 bis SS 2009)		

Appelrath, Hans-Jürgen und Ludewig, Jochen, "Skriptum Informatik - eine konventionelle Einführung", Verlag der Fachvereine Zürich und B.G. Teubner Stuttgart, 4. Auflage 1999

Nagl., M., "Softwaretechnik mit Ada 95. Entwicklung großer Systeme.", Vieweg-Verlag, Wiesbaden 1999

Barnes, J.G.P., "Programming in Ada 95", 2. Auflage, Addison-Wesley 1998

15. Lehrveranstaltungen und -formen:   
 • 102801 Vorlesung Programmierung und Softwareentwicklung   
 • 102802 Übung Programmierung und Softwareentwicklung

16. Abschaetzung Arbeitsaufwand:   
 Präsenzzeit: 63 Stunden   
 Vor-/Nachbearbeitungszeit: 187 Stunden   
 Prüfungsvorbereitung: 20 Stunden

17a. Studienleistung:   
 Studienleistung: Übungsschein, Vor. 3 mal vortragen in den   
 Übungen und mindestens 50% der Übungspunkte erwerben,   
 Teilnahme an den Zwischenklausuren.

17b. Prüfungsleistungen:   
 Modulprüfung: Schriftliche Prüfung von 120 Minuten Dauer

18. Grundlage für ... :

19. Medienform:

20. Prüfungsnummer/n und -name:   
 10281 Programmierung und Software-Entwicklung

21. Angeboten von:

22. Zuordnung zu weiteren Curricula:   
 B.Sc. Softwaretechnik, 1. Semester   
     → Basismodule   
 B.Sc. Wirtschaftsinformatik, 1. Semester   
     → Informatik (B 1)   
 B.Sc. Mechatronik, 1. Semester   
     → Kernmodule   
 B.Sc. Maschinelle Sprachverarbeitung, 1. Semester   
     → Basismodule   
 B.Sc. Technikpädagogik, 1. Semester   
     → Hauptfach Informatik   
     → Basismodule Informatik   
 B.Sc. Technikpädagogik, 1. Semester   
     → Wahlpflichtfach   
     → Informatik   
 M.Sc. Technikpädagogik, 1. Semester   
     → Wahlpflichtfach B   
     → Affines Wahlpflichtfach Informatik   
     → Affines Wahlpflichtfach Informatik Basismodule   
 B.Sc. Simulation Technology, 1. Semester   
     → Grundstudium   
 ohne Absch Lehramt-Pool, 1. Semester   
     → Pflichtmodule

## Modul: 10940 Theoretische Grundlagen der Informatik

2. Modulkürzel:	050420005	5. Moduldauer:	2 Semester
3. Leistungspunkte:	12.0 LP	6. Turnus:	jedes 2. Semester, WiSe
4. SWS:	8.0	7. Sprache:	Deutsch
8. Modulverantwortlicher:	Volker Diekert		
9. Dozenten:	<ul style="list-style-type: none"> <li>• Ulrich Hertrampf</li> <li>• Volker Diekert</li> </ul>		
10. Zuordnung zum Curriculum:	B.Sc. Informatik, 1. Semester → Basismodule		
11. Voraussetzungen:	Keine		
12. Lernziele:	<ul style="list-style-type: none"> <li>• Logik und Diskrete Strukturen: Die Studierenden haben die grundsätzlichen Kenntnisse in Logik und Diskreter Mathematik erworben, wie sie in den weiteren Grundvorlesungen der Informatik in verschiedenen Bereichen benötigt werden.</li> <li>• Automaten und Formale Sprachen: Die Studierenden beherrschen wichtige theoretische Grundlagen der Informatik, insbesondere die Theorie und Algorithmik endlicher Automaten. Hierzu gehört das Kennenlernen, Einordnung und Trennung der Chomskyschen Sprachklassen.</li> </ul>		
13. Inhalt:	<ul style="list-style-type: none"> <li>• Logik und Diskrete Strukturen: Einführung in die Aussagenlogik; formale Sprache; Semantik (Wahrheitswerte); Syntax (Axiome und Schlussregeln); Normalformen; Hornformeln; aussagenlogische Resolution; Korrektheit und Vollständigkeit für die Aussagenlogik; Einführung in die Prädikatenlogik 1. Stufe; formale Sprache; Semantik und Syntax; Normalformen; Herbrand-Theorie; prädikatenlogische Resolution; Kombinatorik, Graphen, elementare Zahlentheorie: Rechnen mit Restklassen, endliche Körper, RSA-Verfahren.</li> <li>• Automaten und Formale Sprachen: Deterministische- bzw. nichtdeterministische endliche Automaten, reguläre Ausdrücke, Minimierung endlicher Automaten, Iterationslemmata für reguläre und kontextfreie Sprachen, Normalformen, Kellerautomaten, Lösen des Wortproblems kontextfreier Sprachen mit dem CYK-Algorithmus, linear beschränkte Automaten, kontextsensitive Grammatiken, Typ 0-Grammatiken und Turingmaschinen.</li> </ul>		
14. Literatur:	<ul style="list-style-type: none"> <li>• John Hopcroft, Jeffrey Ullman, Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie, 1988</li> <li>• Uwe Schöning, Theoretische Informatik - kurzgefasst, 1999</li> </ul>		
15. Lehrveranstaltungen und -formen:	<ul style="list-style-type: none"> <li>• 109401 Vorlesung Logik und Diskrete Strukturen</li> <li>• 109402 Übung Logik und Diskrete Strukturen</li> <li>• 109403 Vorlesung Automaten und Formale Sprachen</li> <li>• 109404 Übung Automaten und Formale Sprachen</li> </ul>		
16. Abschaetzung Arbeitsaufwand:	Präsenzzeit: 84 Stunden		

---

Nachbearbeitungszeit: 276 Stunden

---

17a. Studienleistung: Prüfungsvorleistung: Übungsschein.

---

17b. Prüfungsleistungen: Schriftliche Prüfung von 120 Minuten Dauer.

---

18. Grundlage für ... :

---

19. Medienform:

---

20. Prüfungsnummer/n und -name: 10941 Theoretische Grundlagen der Informatik

---

21. Angeboten von:

---

22. Zuordnung zu weiteren Curricula:

- B.Sc. Softwaretechnik, 1. Semester
  - Basismodule
- B.Sc. Maschinelle Sprachverarbeitung, 1. Semester
  - Basismodule
- B.Sc. Technikpädagogik, 1. Semester
  - Hauptfach Informatik
  - Kernmodule Informatik
- B.Sc. Technikpädagogik, 1. Semester
  - Wahlpflichtfach
  - Informatik
- M.Sc. Technikpädagogik, 1. Semester
  - Wahlpflichtfach B
  - Affines Wahlpflichtfach Informatik
  - Affines Wahlpflichtfach Informatik Kernmodule
- ohne Absch Lehramt-Pool, 1. Semester
  - Pflichtmodule

---