

## LEHRVERANSTALTUNGSBESCHREIBUNG

### 1. Angaben zum Programm

1.1 Hochschuleinrichtung	<b>Babes-Bolyai Universität, Cluj-Napoca</b>
1.2 Fakultät	Mathematik und Informatik
1.3 Department	Informatik
1.4 Fachgebiet	Informatik
1.5 Studienform	Bachelor
1.6 Studiengang / Qualifikation	Informatik

### 2. Angaben zum Studienfach

2.1 LV-Bezeichnung	Fortgeschrittene Programmierungsmethoden						
2.2 Lehrverantwortlicher – Vorlesung	Lect. dr. Cătălin Rusu						
2.3 Lehrverantwortlicher – Seminar	Lect. dr. Cătălin Rusu						
2.4 Studienjahr	2	2.5 Semester	3	2.6 Prüfungsform	P	2.7 Art der LV	Verpflichtend

### 3. Geschätzter Workload in Stunden

3.1 SWS	5	von denen: 3.2 Vorlesung	2	3.3 Seminar/Übung	1+2
3.4 Gesamte Stundenanzahl im Lehrplan	70	von denen: 3.5 Vorlesung	28	3.6 Seminar/Übung	14+28
Verteilung der Studienzeit:					Std.
Studium nach Handbücher, Kursbuch, Bibliographie und Mitschriften					20
Zusätzliche Vorbereitung in der Bibliothek, auf elektronischen Fachplattformen und durch Feldforschung					10
Vorbereitung von Seminaren/Übungen, Präsentationen, Referate, Portfolios und Essays					30
Tutorien					10
Prüfungen					10
Andere Tätigkeiten: .....					-
3.7 Gesamtstundenanzahl Selbststudium	80				
3.8 Gesamtstundenanzahl / Semester	150				
3.9 Leistungspunkte	6				

### 4. Voraussetzungen (falls zutreffend)

4.1 curricular	<ul style="list-style-type: none"> <li>• Objektorientierte Programmierung</li> <li>• Datenstrukturen und Algorithmen</li> </ul>
4.2 kompetenzbezogen	Grundlegende Begriffe der objektorientierter Programmierung.

## 5. Bedingungen (falls zutreffend)

5.1 zur Durchführung der Vorlesung	<ul style="list-style-type: none"><li>• Vorlesungsraum, Beamer, Laptop</li></ul>
5.2 zur Durchführung des Seminars / der Übung	<ul style="list-style-type: none"><li>• Labor ausgestattet mit JAVA</li></ul>

## 6. Spezifische erworbene Kompetenzen

<b>Berufliche Kompetenzen</b>	<p>K1.1 Geeignete Beschreibung der Paradigmen der Programmierung und der spezifischen Sprachmechanismen, sowie die Identifizierung der Differenzen zwischen semantischen und syntaktischen Aspekten</p> <p>K1.2 Erklärung existierender Softwareanwendungen auf verschiedenen Niveaus (Architektur, Pakete, Klassen, Methoden), anhand geeigneter Anwendung der Grundkenntnisse</p> <p>K1.3 Entwickeln von geeigneten Quellcodes und unitäres Testen von Komponenten in einer bekannten Programmiersprache, anhand gegebener Entwurfsspezifikationen</p> <p>K1.4 Testen der Anwendungen anhand von Testplänen</p> <p>K1.5 Entwurf von Programmeinheiten und Verfassung der geeigneten Dokumentationen</p>
<b>Transversale Kompetenzen</b>	<p>TK1 Anwendung der Regeln für gut organisierte und effiziente Arbeit, für verantwortungsvolle Einstellungen gegenüber der Didaktik und der Wissenschaft, für kreative Förderung des eigenen Potentials, mit Rücksicht auf die Prinzipien und Normen der professionellen Ethik</p> <p>TK3 Anwendung von effizienten Methoden und Techniken für Lernen, Informieren und Recherchieren, für das Entwickeln der Kapazitäten der praktischen Umsetzung der Kenntnisse, der Anpassung an die Bedürfnisse einer dynamischen Gesellschaft, der Kommunikation in rumänischer Sprache und in einer internationalen Verkehrssprache</p>

## 7. Ziele (entsprechend der erworbenen Kompetenzen)

7.1 Allgemeine Ziele der Lehrveranstaltung	<ul style="list-style-type: none"><li>• Kenntnisse im Umgang mit Datenmodellen, insbesondere die der objektorientierter Programmierung.</li><li>• Anwenden der Schablone</li><li>• Anwenden von frameworks</li></ul>
7.2 Spezifische Ziele der Lehrveranstaltung	<ul style="list-style-type: none"><li>• Analyse und Design von Softwaresysteme</li><li>• Anlernen grundlegender Kenntnisse über UML</li></ul>

	<ul style="list-style-type: none"> <li>• Aneignen von Schablonen.</li> <li>• Java</li> <li>• C#</li> </ul>
--	--

## 8. Inhalt

8.1 Vorlesung	Lehr- und Lernmethode	Anmerkungen
1. Einführung in Java I	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
2. Einführung in Java II	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
3. IO. Collections. Generische Typen	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
4. Funktionale Programmierung in JAVA. JAVA8. Streams	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
5. GUI. JAVA FX.	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
6. Treads.	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
7-8. XML: schema, documents. JSON	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
9. GUI (Teil 2): FXML, CSS. Metaprogramming: reflection, serialization	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
10. Webprogrammierung. JSP	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
11. Einführung in C#	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
12. LINQ	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
13-14. GRASP und GoF Schablone. Das objektorientierte Design. UML	Vortrag, Unterrichtsgespräch, Erklärungen, Beispiele	
Literatur		

1. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. The Java™ Language Specification Java SE 7 Edition.
2. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006
3. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994
4. Joseph Albahari and Ben Albahari, C# 4.0 in a Nutshell, Fourth Edition, O’Reilly, 2010
5. \*\*\*, Microsoft Developer Network, Microsoft Inc., <http://msdn.microsoft.com/>
6. \*\*\*, The Java Tutorial, 2013. <http://download.oracle.com/javase/tutorial/>

**Auf Deutsch:**

1. Gregor Fischer, Jurgen Wolff von Gudenberg, Programmieren in Java, Springer, Berlin, Heidelberg, New York, 2005.
2. Detlef Sesse, Grundkurs Programmieren in Java 1, Hanser Verlag, Muenchen, Wien, 2007.

8.2 Seminar / Übung / Labor	Lehr- und Lernmethode	Anmerkungen
1. Einfache Übungen	Beispiele, Diskussionen, Debatte, Erklärungen	
2. Java Projekt: Collections, Generics	Beispiele, Diskussionen, Debatte, Erklärungen	
3. Java Projekt: Generics	Beispiele, Diskussionen, Debatte, Erklärungen	
4. Java Projekt: IO	Beispiele, Diskussionen, Debatte, Erklärungen	
5. Java Projekt: Functional programming	Beispiele, Diskussionen, Debatte, Erklärungen	
6. Java Projekt: GUI	Beispiele, Diskussionen, Debatte, Erklärungen	
7. Java Projekt: concurrency	Beispiele, Diskussionen, Debatte, Erklärungen	
8. C# Projekt	Beispiele, Diskussionen, Debatte, Erklärungen	

**Literatur**

1. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. The Java™ Language Specification Java SE 7 Edition.
2. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006

3. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994
4. Joseph Albahari and Ben Albahari, C# 4.0 in a Nutshell, Fourth Edition, O'Reilly, 2010
5. \*\*\*, Microsoft Developer Network, Microsoft Inc., <http://msdn.microsoft.com/>
6. \*\*\*, The Java Tutorial, 2013. <http://download.oracle.com/javase/tutorial/>

**9. Verbindung der Inhalte mit den Erwartungen der Wissensgemeinschaft, der Berufsverbände und der für den Fachbereich repräsentativen Arbeitgeber**

Diese Vorlesung wird an international bekannten Universitäten im Fachgebiet Informatik angeboten.

Der Inhalt der Vorlesung ist wichtig für die Softwarefirmen und entspricht der ACM Richtlinien.

**10. Prüfungsform**

Veranstaltungsart	10.1 Evaluationskriterien	10.2 Evaluationsmethoden	10.3 Anteil an der Gesamtnote
10.4 Vorlesung	Angeeignete Kenntnisse	schriftliche Abschlussarbeit	40%
10.5 Seminar / Übung	Programmieren	Praktischer Test	30%
	Laborarbeiten	Dokumentation, Diskussion	30%
<b>10.6 Minimale Leistungsstandards</b>			
<p>Note jeder Übung (Labor) soll größer als 5 sein. Für das Bestehen der praktischen Prüfung muss die Mindestnote 5 erreicht werden. Für das Bestehen der schriftlichen Prüfung muss die Mindestnote 5 erzielt werden.</p> <p>Nur die Endnote wird auf ganze Zahl gerundet.</p> <p>Erforderliche Anwesenheit beim: Seminar: 75%; Labor: 90%.</p>			

Ausgefüllt am:

Vorlesungsverantwortlicher

Seminarverantwortlicher

Lect. Dr. Cătălin Rusu

Lect. Dr. Cătălin Rusu

Genehmigt im Department am:

Departmentdirektor

Prof. Dr. Anca Andreica