

# LEHRVERANSTALTUNGSBESCHREIBUNG

## 1. Angaben zum Programm

1.1 Hochschuleinrichtung	Babes-Bolyai Universität
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	Logische und Funktionale Programmierung						
2.2 Lehrverantwortlicher – Vorlesung	Conf. Dr. Christian Sacarea						
2.3 Lehrverantwortlicher – Seminar	Conf. Dr. Christian Sacarea						
2.4 Studienjahr	2	2.5 Semester	3	2.6 Prüfungsform	Kolloquium	2.7 Art der LV	Pflichtfach
2.8 Modulnummer	MLG5009						

## 3. Geschätzter Workload in Stunden

3.1 SWS	4	von denen: 3.2 Vorlesung	2	3.3 Seminar/Übung	2
3.4 Gesamte Stundenanzahl im Lehrplan	42	von denen: 3.5 Vorlesung	28	3.6 Seminar/Übung	28
Verteilung der Studienzeit:					Std.
Studium nach Handbücher, Kursbuch, Bibliographie und Mitschriften					25
Zusätzliche Vorbereitung in der Bibliothek, auf elektronischen Fachplattformen und durch Feldforschung					17
Vorbereitung von Seminaren/Übungen, Präsentationen, Referate, Portfolios und Essays					30
Tutorien					18
Prüfungen					18
Andere Tätigkeiten: .....					-
3.7 Gesamtstundenanzahl Selbststudium	108				
3.8 Gesamtstundenanzahl / Semester	150				
3.9 Leistungspunkte	6				

## 4. Voraussetzungen (falls zutreffend)

4.1 curricular	•
4.2 kompetenzbezogen	•

## 5. Bedingungen (falls zutreffend)

5.1 zur Durchführung der Vorlesung	<ul style="list-style-type: none"><li>• Videoprojektor</li></ul>
5.2 zur Durchführung des Seminars / der Übung	<ul style="list-style-type: none"><li>• GCLisp, CLisp, TurboProlog</li></ul>

## 6. Spezifische erworbene Kompetenzen

<b>Berufliche Kompetenzen</b>	<ul style="list-style-type: none"> <li>• die Kenntnisse von Konzepten der deklarativen Programmierung</li> <li>• das Aneignen der Paradigmen der logischen und funktionalen Programmierung</li> <li>• die Vertrautheit mit rekursiver Programmierung</li> <li>• die Fähigkeit einfache Programme in CLisp und Prolog zu schreiben</li> </ul>
<b>Transversale Kompetenzen</b>	<ul style="list-style-type: none"> <li>• die Fähigkeit die deklarative Programmierung für Probleme der realen Welt einzusetzen</li> <li>• die verantwortliche Lösung der Aufgaben</li> <li>• die Anwendung der strengen und effizienten Arbeitsregeln</li> <li>• die Erfüllung der Grundsätzen und Regeln der Berufsethik</li> </ul>

## 7. Ziele (entsprechend der erworbenen Kompetenzen)

7.1 Allgemeine Ziele der Lehrveranstaltung	<ul style="list-style-type: none"> <li>• die Vertrautheit mit deklarativer Programmierung</li> </ul>
7.2 Spezifische Ziele der Lehrveranstaltung	<ul style="list-style-type: none"> <li>• die Einführung einer Programmiersprache für jedes Paradigma (CLisp, Prolog)</li> <li>• die Idee der Verwendung dieser Paradigmen für die Bedürfnisse der Softwareprogramme</li> <li>• die Grundlagen für nachfolgende fortgeschrittene Programmierkurse anzubieten</li> </ul>

## 8. Inhalt

8.1 Vorlesung	Lehr- und Lernmethode	Anmerkungen
<i>Logische Programmierung. PROLOG</i>		
1. Rekursion und Programmiersprachen. Imperative Programmierung und deklarative Programmierung. Einführung. Rekursion. Beispiele.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
2. Grundlagen der Prolog-Programmierung. Prolog Fakten und Regeln. Fragen. Kontrollstrukturen in Prolog. Variable und zusammengesetzte Terme. Anonyme Variable. Planungssysteme. Die Teile eines Prolog-programms. Beispiele.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
3. Das Prolog-programm. Vordefinierte Domäne. Interne und externe Fragen. Multiplizität der Prädikate. Das IF Symbol und der IF Befehl. Arithmetik und Vergleiche. Eingabe und Ausgabe. Zeichenketten	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
4. Backtracking. Die "Fail" und "!" Prädikate. Das "Not" Prädikat. Listen. Rekursion.	Beschreibung, Erklärungen, Unterrichtsgespräch,	

Beispiele.	Vorführung	
5. Zusammengesetzteobjekte und Funktoren (bzw "Funktionsobjekte"). Die Vereinigung der Zusammengesetzte Objekte. Vergleich von Zusammengesetzte Objekte. Backtracking mit Zyklen. Beispiele der rekursiven Programmierung. Der StackFrame. Tail Recursion und Optimierung.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
6. Rekursive Datenstrukturen. Bäume als Datenstruktur. Suche in Bäumen. Suchbäume. Die intern Datenbank der Prolog.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
7. Prüfung		
<i>Funktionale Programmierung. LISP</i>		
8. Die Bedeutung der funktionalen Programmierung als ein neues Programmierparadigma. Einführung in Lisp. Grundelemente von Lisp. Dynamische Datenstrukturen. Semantische und syntaktische Regeln. Klassifikation der Lisp-Funktionen. Primitive Funktionen.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
9. Lisp Prädikate. Logische und arithmetische Funktionen. Benutzerdefinierte Funktionen. Beispiele.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
10. Die Verwaltung der Symbole. Weitere Listenbefehle. OBLIST, ALIST. Weitere interessante Funktionen. Beispiele.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
11. Die Eval-Funktion. Die FUNCALL und APPLY Funktionen. Lambda-Funktionen. Label-Funktionen. Beispiele.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
12. Generators. MAP-Funktionen. Beispiele.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
13. Weitere Grundelemente.	Beschreibung, Erklärungen, Unterrichtsgespräch, Vorführung	
14. Prüfung		
Literatur 1. CZIBULA G., POP H.F., Elemente avansate de programare in Lisp si Prolog. Aplicatii in Inteligenta Artificiala, Editura Albastra, Cluj-Napoca, 2012 2. POP H.F., SERBAN G., Programare in Inteligenta Artificiala - Lisp si Prolog, Editura Albastra, Cluj-Napoca, 2003 3. <a href="http://www.ifcomputer.com/PrologCourse">http://www.ifcomputer.com/PrologCourse</a> , Lecture on Prolog 4. <a href="http://www.lpa.co.uk">http://www.lpa.co.uk</a> , Logic Programming 5. FIELD A., Functional Programming, Addison Wesley, New York, 1988. 6. WINSTON P.H., Lisp, Addison Wesley, New York, 2nd edition, 1984. 7. GOOS, G., ZIMMERMANN, W., Vorlesungen über Informatik, Band 1, Grundlagen und funktionales Programmieren, Springer, 2006. 8. LIPPE, W-M., Funktionale und Applikative Programmierung, Springer 2009.		
8.2 Seminar / Übung	Lehr- und Lernmethode	Anmerkungen
1. Rekursion	Erklärungen,	2 Stunden jeden 2 Woche

	Unterrichtsgespräch, Modellierung	
2. Listen	Erklärungen, Unterrichtsgespräch, Modellierung	
3. Bäume	Erklärungen, Unterrichtsgespräch, Modellierung	
4. Backtracking	Erklärungen, Unterrichtsgespräch, Modellierung	
4. Praktische Klausur		1 Stunde
5. Rekursive Programmierung	Erklärungen, Unterrichtsgespräch, Modellierung	
6. MAP-Funktionen	Erklärungen, Unterrichtsgespräch, Modellierung	
7. Iterative Programmierung	Erklärungen, Unterrichtsgespräch, Modellierung	
7. Praktische Klausur		1 Stunde
Literatur		
1. CZIBULA G., POP H.F., Elemente avansate de programare in Lisp si Prolog. Aplicatii in Inteligenta Artificiala, Editura Albastra, Cluj-Napoca, 2012 2. Documentatia produselor: Gold Common Lisp 1.01 si 4.30, XLisp, Free Lisp. 3. Documentatia produselor: Turbo Prolog 2.0, Logic Explorer, Sicstus Prolog. 4. <a href="http://www.swi-prolog.org/">http://www.swi-prolog.org/</a>		

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

<ul style="list-style-type: none"> <li>• Der Kurs existiert in der Mehrzahl der rumänischen und ausländischen Universitäten.</li> <li>• Der Kursinhalt bietet die notwendigen Lisp und Prolog Programmierkenntnisse für eine eventuelle Arbeitsstelle.</li> </ul>
---

### 10. Prüfungsform

Veranstaltungsart	10.1 Evaluationskriterien	10.2 Evaluationsmethoden	10.3 Anteil an der Gesamtnote
10.4 Vorlesung	Die Richtigkeit und die Vollständigkeit der erworbenen Kenntnisse	Schriftliche Prolog-Klausur (während des Semesters)	30%
		Schriftliche Lisp-Klausur (während des Semesters)	30%
10.5 Seminar / Übung	- Die Umsetzung der Aufgaben in Lisp und Prolog - Die Ausarbeitung der Dokumentation - Die Einhaltung der Frist	die Softwaredokumentation	10%

	für die Aufgaben		
		Praktische Prolog-Klausur (1 Stunde)	15%
		Praktische Lisp-Klausur (1 Stunde)	15%
<b>10.6 Minimale Leistungsstandards</b>			
<ul style="list-style-type: none"> <li>• Kenntnisse der Grundlagen. Die Lösung einer einfache Aufgabe in einer deklarative Programmiersprache. Jeder Student muss einen akzeptablen Kenntnisstand beweisen</li> <li>• Min 4 auf jeder Klausur. Min 5 auf jeder Übung. Die Gesamtnote min 5 auf einer Skala von 1 bis 10.</li> </ul>			

Ausgefüllt am:

Vorlesungsverantwortlicher

Seminarverantwortlicher

Conf. Dr. Christian Sacarea

Conf. Dr. Christian Sacarea

Genehmigt im Department am:

Departmentdirektor

Prof. Dr. Anca Andreica