

LEHRVERANSTALTUNGSBESCHREIBUNG

1. Angaben zum Programm

1.1 Hochschuleinrichtung	Babes-Bolyai Universität, Cluj-Napoca
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	RECHNERARCHITEKTUR						
2.2 Lehrverantwortlicher – Vorlesung	Prof. Dr. Anca Andreica						
2.3 Lehrverantwortlicher – Seminar	Prof. Dr. Anca Andreica						
2.4 Studienjahr	1	2.5 Semester	1	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/Labor	1+2
3.4 Gesamte Stundenanzahl im Lehrplan	70	von denen: 3.5 Vorlesung	28	3.6 Seminar/Übung	42
Verteilung der Studienzeit:					Std.
Studium nach Handbücher, Kursbuch, Bibliographie und Mitschriften					25
Zusätzliche Vorbereitung in der Bibliothek, auf elektronischen Fachplattformen und durch Feldforschung					10
Vorbereitung von Seminaren/Übungen, Präsentationen, Referate, Portfolios und Essays					25
Tutorien					5
Prüfungen					15
Andere Tätigkeiten:					
3.7 Gesamtstundenanzahl Selbststudium	80				
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	•
5.2 zur Durchführung des Seminars / der Übung	• Computerraum

6. Spezifische erworbene Kompetenzen

Berufliche Kompetenzen	<p>K 6.1 Identifizierung der Konzepte und Modelle für Rechnersysteme und Rechnernetze</p> <p>K 6.2 Identifizierung und Erklärung der Basisarchitektur für die Verwaltung vernetzter Rechnersysteme</p>
Transversale Kompetenzen	<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"> • Diese Veranstaltung soll die Studierenden befähigen, die inhärenten Prinzipien aktueller Prozessoren zu verstehen, um diese bei der Programmierung angemessen nutzen zu können. • Vermittlung der notwendigen Grundlagen zur maschinennahen Programmierung anhand der Assemblerprogrammierung, der Interaktion mit verschiedener Hardware sowie der Einbindung von Assemblermodule in höhere Programmiersprachen.
7.2 Spezifische Ziele der Lehrveranstaltung	<ul style="list-style-type: none"> • Einführung in der Assembler Programmierung, sowie das Verstehen grundlegender Abläufe in allgemeinen Rechen- und Steuereinheiten moderner Rechensysteme. • Einführung in der Architektur der Rechnersysteme, insbesondere auf die 80x86 Maschinen. • Verstehen prinzipielle Methoden der Leistungssteigerung hinsichtlich Speicherzugriff, Verbindungsstruktur, Ein-/Ausgabe und Befehlsabarbeitung vorgestellt, sowie den Einfluss der von Neumann Architektur auf die modernen Programmiersprachen.

8. Inhalt

8.1 Vorlesung	Lehr- und Lernmethode	Anmerkungen
1. Darstellung von Daten	Vortrag, Erklärung, Debatte, praktische Beispiele	
2. Darstellung von Daten	Vortrag, Erklärung, Debatte, praktische Beispiele	
3. Die Architektur des 80x86 Mikroprozessors	Vortrag, Erklärung, Debatte, praktische Beispiele	
4. Die Architektur des 80x86 Mikroprozessors	Vortrag, Erklärung, Debatte, praktische Beispiele	

5. Assembler-Anweisungen und -Operanden	Vortrag, Erklärung, Debatte, praktische Beispiele	
6. Assembler-Anweisungen und -Operanden	Vortrag, Erklärung, Debatte, praktische Beispiele	
7. Assembler-Anweisungen und -Operanden	Vortrag, Erklärung, Debatte, praktische Beispiele	
8. Assemblerbefehle	Vortrag, Erklärung, Debatte, praktische Beispiele	
9. Assemblerbefehle	Vortrag, Erklärung, Debatte, praktische Beispiele	
10. Assemblerbefehle	Vortrag, Erklärung, Debatte, praktische Beispiele	
11. Unterbrechungen	Vortrag, Erklärung, Debatte, praktische Beispiele	
12. Multi-Modul-Programmierung	Vortrag, Erklärung, Debatte, praktische Beispiele	
13. Multi-Modul-Programmierung	Vortrag, Erklärung, Debatte, praktische Beispiele	
14. Low level Programmierung in höheren Programmiersprachen	Vortrag, Erklärung, Debatte, praktische Beispiele	

Literatur

1. Axel Bottcher, Rechneraufbau und Rechnerarchitektur, Springer Verlag, Berlin, Heidelberg, New York, 2006.
2. Oberschelb Walter, Vossen Gottfried, Rechneraufbau und Rechnerstrukturen, Oldenburg, Muenchen, Wien, 2000.
3. R. Richter, P. Sander, W. Stucky, Der Rechner als System, Organisation, Daten, Programme, Teubner, Stuttgart, 1997.
4. Christian Siemers, Rechnerarchitektur I/II, http://www.in.tu-clausthal.de/uploads/media/Rechnerarchitektur_Skript_02.pdf
5. P. Marwedel, Rechnerarchitektur, <https://eldorado.tu-dortmund.de/bitstream/2003/20362/1/rechner.pdf>
6. Ch. Martin, Rechnerarchitekturen : CPUs, Systeme, Software - Schnittstellen, Fachbuchverlag, Leipzig, 2002
7. Al. Vancea, F. Boian, D. Bufnea, A. Gog, A. Darabant, A. Sabau – Arhitectura calculatoarelor. Limbajul de asamblare 80x86., Editura Risoprint, Cluj-Napoca, 2005.
8. A. Gog, A. Sabau, D. Bufnea, A. Sterca, A. Darabant, Al. Vancea – Programarea în limbaj de asamblare 80x86. Exemple si aplicatii., Editura Risoprint, Cluj-Napoca, 2005.
9. Randal Hyde – The Art of Assembly Programming, No Starch Press, 2003.
(<http://homepage.mac.com/randyhyde/webster.cs.ucr.edu/www.artofasm.com/DOS/index.html>)
10. Boian F. M. De la aritmetica la calculatoare. Ed. Presa Universitara Clujeana, Cluj, 1996
11. Boian F.M. Vancea A. Arhitectura calculatoarelor, suport de curs. Facultatea de Matematica si Informatica, Centrul de Formare Continua si Invatamânt la Distanta., Ed. Centrului de Formare Continua si Invatamânt la Distanta, Cluj, 2002

8.2 Seminar / Übung	Lehr- und Lernmethode	Anmerkungen
1. Information und Zahlendarstellung	Beispiele, Diskussionen	<i>Das zweistündige Seminar findet jede zweite Woche statt; Die Laborstunden finden wöchentlich statt und die Inhalte sind mit dem Seminarinhalten synchronisiert.</i>
2 – 3. Assemblerbefehle: Transfer, Umrechnungen, arithmetische Operationen. Von-Neumann-Rechnerarchitektur	Beispiele, Diskussionen	
4. Bitoperationen, logische Bitoperationen	Beispiele, Diskussionen	
5-6. Assembler-Programmierung	Beispiele, Diskussionen	
7. Assembler-Programmierung	Beispiele, Diskussionen	
Literatur		

1. <http://homepages.fh-friedberg.de/euler/wi/skript.pdf>
2. Andrew S. Tanenbaum, Computerarchitektur. Strukturen - Konzepte - Grundlagen Addison-Wesley Verlag, 2005
3. Al. Vancea, F. Boian, D. Bufnea, A. Gog, A. Darabant, A. Sabau – Arhitectura calculatoarelor. Limbajul de asamblare 80x86., Editura Risoprint, Cluj-Napoca, 2005.
4. A. Gog, A. Sabau, D. Bufnea, A. Sterca, A. Darabant, Al. Vancea – Programarea în limbaj de asamblare 80x86. Exemple si aplicatii., Editura Risoprint, Cluj-Napoca, 2005.
5. Randal Hyde – The Art of Assembly Programming, No Starch Press, 2003.
(<http://homepage.mac.com/randyhyde/webster.cs.ucr.edu/www.artofasm.com/DOS/index.html>)

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 des Kurses gilt als wichtiger Teil der Programmierkenntnisse der Informatiker in Software-Unternehmen.

10. Prüfungsform

Veranstaltungsart	10.1 Evaluationskriterien	10.2 Evaluationsmethoden	10.3 Anteil an der Gesamtnote
10.4 Vorlesung	Grundkenntnisse der Prinzipien der Rechnerarchitektur	Schriftliche Prüfung	50%
	Anwendung dieser Konzepte in der Lösung von Aufgaben		
10.5 Seminar / Übung	Durchführen von Assembler-Programmen	Labor Hausaufgaben	20%
		Seminar	10%
		Praktische Prüfung	20%
10.6 Minimale Leistungsstandards			
Für das Bestehen der Prüfung muss die Mindestnote 5 erzielt werden (bei der schriftlichen, bzw. praktischen Prüfung, und beim Lösen der Hausaufgaben).			

Ausgefüllt am:

Vorlesungsverantwortlicher

Seminarverantwortlicher

Prof. Dr. Anca Andreica

Prof. Dr. Anca Andreica

Genehmigt im Department am:

Departmentdirektor

Prof. Dr. Anca Andreica