Modulbeschreibung
39-Inf-8_a Rechnerarchitektur

Technische Fakultät

Version vom 29.11.2026

Dieses Modulhandbuch gibt den derzeitigen Stand wieder und kann Änderungen unterliegen. Aktuelle Informationen und den jeweils letzten Stand dieses Dokuments finden Sie im Internet über die Seite

https://ekvv.uni-bielefeld.de/sinfo/publ/modul/420556013

Die jeweils aktuellen und gültigen Regelungen im Modulhandbuch sind verbindlich und konkretisieren die im Verkündungsblatt der Universität Bielefeld veröffentlichten Fächerspezifischen Bestimmungen.

39-Inf-8_a Rechnerarchitektur

Fakultät

Technische Fakultät

Modulverantwortliche*r

Prof. Dr.-Ing. Ralf Möller

Turnus (Beginn)

Jedes Wintersemester

Leistungspunkte

5 Leistungspunkte

Kompetenzen

Die Studierenden verstehen die Architektur moderner Digitalrechner von Teilaspekten der Realisierung auf der Hardware-Ebene über die Kodierung von Zahlen und Zeichen bis zur "instruction set architecture". Durch die Übungsaufgaben werden die Studierenden befähigt, Assembler-Programme einer (weit verbreiteten) Beispielarchitektur zu analysieren, wodurch sie auch in die Lage versetzt werden, von einem Hochsprachen-Compiler erzeugten Assemblercode zu verstehen und auf Effizienz zu prüfen. Die Studenten sind darüber hinaus in der Lage, selbst Assembler-Programme für die Beispielarchitektur zu schreiben. Dies ermöglicht ihnen die hardwarenahe Programmierung z.B. von Betriebssystem-Komponenten. Die Fähigkeiten zum Analysieren und Erstellen von Assembler-Programmen belegen dabei ein vertieftes Verständnis der Architektur moderner Digitalrechner.

Lehrinhalte

Die Vorlesung Rechnerarchitektur befasst sich mit der Architektur moderner Digitalrechner. Dazu werden Grundlagen zu digitalelektronischen Schaltungen, deren schaltalgebraischer Beschreibung und deren Einsatzbereichen in Digitalrechnern sowie Verfahren zur Kodierung von Zahlen und Zeichen im Digitalrechner vermittelt. Darauf aufbauend wird die Architektur von Digitalrechnern und deren Prozessoren vorgestellt. Eine Einführung in die "instruction set architecture" von Prozessoren erfolgt am Beispiel der x86-64 Architektur und deren Programmierung auf der Ebene von Assembler-Instruktionen. Die Übungen zur Vorlesung vertiefen das Wissen durch theoretische Aufgaben und Aufgaben zur Entwicklung von Assembler-Programmen für x86-64 Prozessoren.

Empfohlene Vorkenntnisse

Erfolgreicher Abschluss der Module 39-Inf-PP und 39-Inf-AD bzw. 39-Inf-1

Notwendige Voraussetzungen

–

Erläuterung zu den Modulelementen

Modulstruktur: 1 bPr [1]

Veranstaltungen

Titel	Art	Turnus	Workload [5]	LP[2]
Rechnerarchitektur	Übung	WiSe	60 h (30 + 30)	2
Rechnerarchitektur	Vorlesung	WiSe	90 h (30 + 60)	3

Prüfungen

Zuordnung Prüfende	Art	Gewichtung	Workload	LP[2]

Modulverantwortliche*r prüft oder bestimmt Prüfer*in *Portfolio mit Abschlussprüfung bestehend aus:* *1) Portfolio von Übungen zu Inhalten der Vorlesung* *Übungsaufgaben oder Programmieraufgaben, die veranstaltungsbezogen gestellt werden (Bestehensgrenze 50% der erzielbaren Punkte). Die Kontrolle der Übungsaufgaben umfasst auch direkte Fragen zu den Lösungsansätzen, die von den Studierenden in den Übungen beantwortet werden müssen. Der*die Lehrende kann ein individuelles Erläutern und Vorführen von Aufgaben verlangen sowie einen Teil der Übungsaufgaben durch Präsenzübungen ersetzen. Die Übungsaufgaben im Rahmen des Portfolios werden in der Regel wöchentlich ausgegeben und dienen dem begleitenden Erlernen selbständiger Umsetzungen der in der Vorlesung vorgestellten Lerninhalte.* *2) einer Abschlussprüfung zur Vorlesung* *Die Abschlussprüfung zu den Inhalten der Vorlesung nimmt Bezug auf die Übungs- oder Programmieraufgaben oder entwickelt sich aus den in den Übungen erlernten Kompetenzen.* *Eine weitergehende Konkretisierung insbesondere zum zeitlichen Umfang der Abschlussprüfung erfolgt in der Beschreibung der Veranstaltung.* *Abschlussklausur (im Umfang von 90-180 Minuten) oder mündliche Abschlussprüfung (im Umfang von 20-40 Minuten) zu den in der Vorlesung vermittelten und in den Übungen erarbeiteten Inhalten.* *Die Klausur kann alternativ als eKlausur, Open Book Klausur oder eOpen Book Klausur geprüft werden. Im Falle von Open Book Klausur und eOpen Book Klausur beträgt der Umfang 120-180 Minuten.* *Alternativ kann ein Essay (im Umfang von ca. A4 Seiten) mit einer stark auf die vermittelten Kenntnisse und Fähigkeiten bezogenen Aufgabenstellung ODER ein Essay (bis zu 4 A4-Seiten als Abschlussbericht) mit einer stark auf die vermittelten Kenntnisse und Fähigkeiten bezogenen Programmieraufgabe von der*dem Lehrenden vorgesehen werden. Es handelt sich um eine reflektive Aufgabenstellung zu Systematik und Zusammenhängen der Lerninhalte oder um eine Auseinandersetzung mit einer Programmieraufgabe zu den erlernten Inhalten.* *Beide Portfolioelemente werden durch eine*n Prüfer*in geprüft. Es erfolgt eine abschließende Gesamtbewertung.*	Portfolio mit Abschlussprüfung	1	-	-

Legende

1	Die Modulstruktur beschreibt die zur Erbringung des Moduls notwendigen Prüfungen und Studienleistungen.
2	LP ist die Abkürzung für Leistungspunkte.
3	Die Zahlen in dieser Spalte sind die Fachsemester, in denen der Beginn des Moduls empfohlen wird. Je nach individueller Studienplanung sind gänzlich andere Studienverläufe möglich und sinnvoll.
4	Erläuterungen zur Bindung: "Pflicht" bedeutet: Dieses Modul muss im Laufe des Studiums verpflichtend absolviert werden; "Wahlpflicht" bedeutet: Dieses Modul gehört einer Anzahl von Modulen an, aus denen unter bestimmten Bedingungen ausgewählt werden kann. Genaueres regeln die "Fächerspezifischen Bestimmungen" (siehe Navigation).
5	Workload (Kontaktzeit + Selbststudium)
SoSe	Sommersemester
WiSe	Wintersemester
SL	Studienleistung
Pr	Prüfung
bPr	Anzahl benotete Modul(teil)prüfungen
uPr	Anzahl unbenotete Modul(teil)prüfungen