Softwaresysteme 1 — SOS 1

Grundlagen von Betriebssystemen

Wolfgang Schröder-Preikschat

Friedrich-Alexander-Universität Erlangen-Nürnberg
Lehrstuhl Informatik 4 (Verteilte Systeme und Betriebssysteme)

www4.informatik.uni-erlangen.de

...---...

Sommersemester 2005

Teil I

Organisation der Lehrveranstaltung

Überblick

Lehrveranstaltungskonzept
- Einordnung
- Studiengänge
- Lernziele und Lehrinhalte
- Voraussetzungen
- Ablauf der Lehrveranstaltung
- Leistungsnachweise
- Kontakt

Säule im Grundstudium Informatik, 2. – 4. Semester

SOS_I	☞	Betriebssysteme	☞	I4
SOS_{II}	☞	Datenbanksysteme	☞	I6
SOS_{III}	☞	Softwaretechnik	☞	I11

Vermittlung der Grundlagen von (großen) Softwaresystemen.

Integrierte Lehrveranstaltung

$$\text{Termine} \left\{ \begin{array}{ll} \text{Vorlesung} & 2 \\ \text{Übung} & 1 \\ \text{Rechner} & 1 \end{array} \right\} \equiv 4 \times 1{,}5 = 6 \text{ Zeitstunden wöchentlich}$$

Vor-/Nacharbeit

- N Stunden wöchentlich: $0 \leq N \leq (162 - X)$
- $X \ll 162$ ist das Zeitstundenäquivalent anderer „Pflichten“

Softwaresysteme ist Kernstoff der Informatik

Diplom

- Informatik, I & K
- Mathematik (Wahlfach Informatik), Technomathematik
- Wirtschaftsinformatik

Bachelor

- Computational Engineering (CE)

Magister

- Linguistische Informatik

Lehramt Informatik

- Gymnasium

Lernziele

Vorgänge innerhalb von Rechensystemen ganzheitlich verstehen

- Zusammenspiel
  - Hardware ↔ Software
  - ⋮
  - Anwendung ↔ Betriebssystem
  - ⋮
  - Treiber ↔ Gerät

  } begreifen

Imperative Systemprogrammierung (in Grundzügen) selbst erleben

- im Kleinen praktizieren ☞ Dienstprogramme
- im Großen erfahren ☞ Betriebssysteme

Lehrinhalte

Vorlesung: Vorstellung und detailierte Behandlung des Lehrstoffs

- Organisation (der Systemsoftware) von Rechensystemen
- Grundlagen von Betriebssystemen
- maschinennahe Programme

Übung: Vertiefung, Besprechung der Übungsaufgaben, Tafelübungen

- Systemprogrammierung in C
- Systemprogramme, -aufrufe, -funktionen von UNIX

Erforderliche Grundkenntnisse

Algorithmik: Grundlagen strukturierter Programmierung

- Datentypen, Kontrollkonstrukte, Prozeduren
- statische und dynamische Datenstrukturen
- „Programmierung im Kleinen“

Technische Informatik: Grundlagen der Rechnerorganisation

- „von Neumann Architektur“
  - Operationsbefehle, Befehlsoperanden, Adressierungsarten
  - Unterbrechungssteuerung (Pegel kontra Flanke)
  - Assemblerprogrammierung
- CPU, DMA, FPU, IRQ, MCU, MMU, NMI, PIC, TLB

Vorlesungsbetrieb und Lehrmaterialien

Vorlesungstermine (ab 25.04.)

Montag	10:15–11:45	H7
Donnerstag	16:15–17:45	H7

Handzettel (engl. *handout*) sind verfügbar wie folgt:

1. `www4.informatik.uni-erlangen.de/Lehre/SS05/V_SOS1`
   - der Verweis führt zu den Folien zum Vorlesungsstoff
2. Gutscheinverkauf zum Bezug von Folienkopien, Kosten 5 EUR
   - die Kopien werden vor der Vorlesung ausgegeben

Fachbegriffe der Informatik (Deutsch ↔ Englisch)

- `http://www.babylonia.ork.uk`

Ergänzende Literatur

- B. W. Kernighan, D. M. Ritchie. *The C Programming Language*. Prentice-Hall, Inc., second edition, 1988. ISBN 0-13-110362-8 (paperback) 0-13-110370-9 (hardback).
- J. Nehmer, P. Sturm. *Systemsoftware: Grundlagen moderner Betriebssysteme*. dpunkt.Verlag GmbH, zweite Edition, 2001. ISBN 3-89864-115-5.
- A. Silberschatz, P. B. Galvin, G. Gagne. *Operating System Concepts*. John Wiley & Sons, Inc., sixth edition, 2001. ISBN 0-471-41643-2.
- A. S. Tanenbaum. *Structured Computer Organization*. Prentice-Hall, Inc., fourth edition, 1999. ISBN 0-13-095990-1.

Übungsbetrieb

Anmeldung nur über WAS (Web-Anmeldesystem):

- `www.was.dienste.uni-erlangen.de/content`
- angegeben werden können Wunschtermine für Tafelübungen
- Beginn des Übungsbetriebs: ab 18.04.

Kompaktkurs zur Einführung in die (System-) Programmiersprache C

- Kurstermine: 11., 13., 14., 18. und 21.04.

Bedeutung von Tafel- und Rechnerübungen

Tafelübungen „*learning by exploring*“

- Besprechung der Übungsaufgaben, Skizzierung von Lösungswegen
- Vertiefung des Vorlesungsstoffes, Klärung offener Fragen

Rechnerübungen „*learning by doing*“

- selbständiges Bearbeiten der Übungsaufgaben am Rechner
- Hilfestellung beim Umgang mit den Entwicklungswerkzeugen
- der Rechner ist kein Tafelersatz, die Betreuung verläuft eher passiv

☞ „Wieso, weshalb, warum? Wer nicht fragt, bleibt dumm!“

Leistungskontrolle

unbenoteter Schein: mindestens 60 % der Gesamtpunkte sind zu erreichen

- obligatorisch für Informatik (Diplom, Lehramt (Gym.)), I & K, CE
- empfohlen für Wirtschaftsinformatik (Diplom)

Ex/Testat: Teil des Scheins, einer von zwei Terminen ist verpflichtend

- 1. Termin: mindestens 40 % der Punkte sind zu erreichen
- 2. Termin: wie bei 1., falls entschuldigt gefehlt, 50 % sonst

studienbegleitende Prüfung: schriftlich (Klausur)

- Zulassung mit Schein und bei bestandener Ex, sofern obligatorisch

Übungsscheinkriterien

Abgabefristen von Übungsaufgaben sind strikt einzuhalten

- der Abgabevorgang wird überwacht durch ein Skriptprogramm
  - prüft die Aufgabenlösungen auf Plausibilität
  - testet auf Abschreiben
- stichprobenartig werden einzelne Aufgaben genauer durchkorrigiert

Scheinvergabe nur bei ausreichender Bearbeitung der Übungsaufgaben

- Plausibilitätsprüfung und Abschreibtest müssen OK signalisiert haben
- die Stichprobenkorrektur muss ein positives Ergebnis geliefert haben
- ggf. entscheidet eine persönliche Rücksprache über Erfolg/Misserfolg

Ex/Testat und Klausur

Ex/Testat: Mehrfachauswahl (engl. *multiple choice*), 45 Minuten

- abgefragt wird Stoff von Vorlesung und Übung (Programmbeispiel)
- Termine im Zeitraum Juni/Juli, Ankündigung eine Woche vorher

Klausur: schriftliche Prüfung, 120 Minuten

- geprüft wird Stoff von Vorlesung und Übung (Programmbeispiele)
- Klausurtermin im Zeitraum September/Oktober

aktive Mitarbeit machen Ex/Testat und Klausur „leicht“

- Programme zwar im Team entwickeln, aber selbst zum Laufen bringen

www4.informatik.uni-erlangen.de/*

Dozenten

- Jürgen Kleinöder (~jklein)
- Wolfgang Schröder-Preikschat (~wosch)

Mitarbeiter

- Stefan Steidl (I5)
- Christian Wawersich (~wawi)

Fragen...

?