

Softwaresysteme (SoS) I

Wolfgang Schröder-Preikschat

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

...--...

Sommersemester 2004

<http://www4.informatik.uni-erlangen.de/~wosch>

Organisation (1)

| | | |
|--------------------|--------------------------|-----|
| SoS _i | ➞ Betriebssysteme | I4 |
| SoS _{ii} | ➞ Datenbanksysteme | I6 |
| SoS _{iii} | ➞ Softwaretechnik | II1 |

Vermittlung der Grundlagen von (großen) Softwaresystemen.

Organisation (2)

SoS_i ➞ Softwaresysteme I

- **integrierte Lehrveranstaltung**

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

- **Vor-/Nacharbeit**

– N Stunden wöchentlich: $0 \leq N \leq (162 - X)^1$

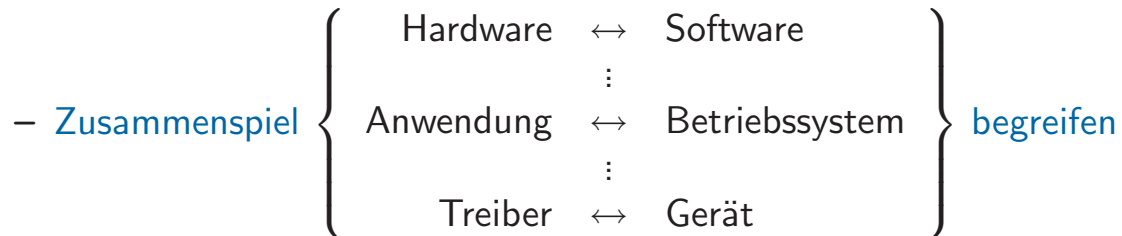
¹ X ist Stellvertreter für den wöchentlichen Arbeitsaufwand (Zeitstunden) anderer „Pflichten“.

Studiengänge

- Diplom
 - Informatik, I & K 2. Semester
 - Mathematik (Wahlfach Informatik), Technomathematik Grundstudium
 - Wirtschaftsinformatik Hauptstudium
- *Bachelor*: Computational Engineering (CE) 2. Semester
- Magister: Linguistische Informatik Grundstudium
- Lehramt Informatik (Gymnasium) Grundstudium

Lernziele

- Vorgänge innerhalb von Rechensystemen *ganzheitlich* erkennen und verstehen



- imperative **Systemprogrammierung** (in Grundzügen) selbst erleben
 - im Kleinen praktizieren ➞ *Dienstprogramme*
 - im Großen erfahren ➞ *Betriebssysteme*

Lehrinhalte

Vorlesung Vorstellung und detaillierte Behandlung des Lehrstoffs

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

Übung Vertiefung, Besprechung der Übungsaufgaben, Tafelübungen

- System $\left\{ \begin{array}{l} \text{programmierung} \\ \text{programme} \\ \text{aufrufe} \\ \text{funktionen} \end{array} \right\} \begin{array}{l} \text{in C} \\ \\ \text{UNIX-ähnlicher Betriebssysteme} \end{array}$

Voraussetzungen

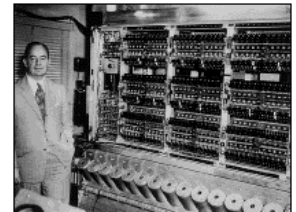
Algorithmik Grundlagen strukturierter Programmierung

- 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



Vorlesung

Termine bis auf **20.05.**, **31.05.** und **10.06.** wie folgt:

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

Skript steht zur Verfügung über zwei Wege:

1. http://www4.informatik.uni-erlangen.de/Lehre/SS04/V_S0S1
 - der Verweis führt zu den Folien zum Vorlesungsstoff
2. Gutscheinverkauf zum Bezug von Folienkopien, Kosten 5 EUR
 - die Kopien werden vor der Vorlesung ausgegeben

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.

Übung (1)

Termine für Tafel- und Rechnerübungen werden wöchentlich angeboten

- Anmeldungen dafür erfolgen nur über WAS (Web-Anmeldesystem):
 - <http://www.was.dienste.uni-erlangen.de/content>
 - angegeben werden können Wunschtermine für Tafelübungen
- Beginn des Übungsbetriebs: KW 18, ab 26.04.

Kompaktkurs C Einführung in die (System-) Programmiersprache C

- erste Kursveranstaltung: Mittwoch den 21.04., 16:00–17:30, im H7

☞ http://www4.informatik.uni-erlangen.de/Lehre/SS04/V_S0S1

Übung (2)

Tafelübungen „*learning by exploring*“

- Besprechung der Übungsaufgaben, Skizzierung möglicher Lösungswege
- Vertiefung des Vorlesungsstoffes, Klärung offener Fragen

Rechnerübungen „*learning by doing*“

- selbständiges Bearbeiten der Übungsaufgaben am Rechner
 - Lösungen/Programme werden nicht „*on line*“ entwickelt
 - der Rechner ist kein Tafelersatz, die Betreuung verläuft eher passiv
- Hilfestellung wird geleistet beim Umgang mit den Entwicklungswerkzeugen



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

Leistungsnachweis

unbenoteter Schein

- Linguistische Informatik (Magister, C-Kompaktkurs)
- bzw. als „**Prüfungsvoraussetzung**“ für die . . .

studienbegleitende Prüfung (Klausur)

| | | |
|------------|-------------------|---|
| Schein ist | Pflicht | <ul style="list-style-type: none">Informatik, I & K, CEInformatik Lehramt (Gymnasium) |
| | Empfehlung | <ul style="list-style-type: none">Wirtschaftsinformatik (Diplom)Techno-/Mathematik (Diplom, Wahlfach Inf.) |

Scheinkriterien

- Übungsaufgaben sind rechtzeitig abzugeben: die Frist ist strikt einzuhalten²
 - der Abgabevorgang wird überwacht durch ein Skriptprogramm !
 - ☞ prüft die Aufgabenlösungen auf Plausibilität
 - ☞ testet auf Abschreiben
 - stichprobenartig werden einzelne Aufgaben genauer durchkorrigiert !
- die Scheinvergabe erfolgt bei ausreichender Bearbeitung der Übungsaufgaben
 - Plausibilitätsprüfung und Abschreibetest müssen OK signalisiert haben
 - die Stichprobenkorrektur muss ein positives Ergebnis geliefert haben
 - ggf. entscheidet eine persönliche Rücksprache über Erfolg/Misserfolg

²Nur in begründeten Notfällen wird eine Fristüberschreitung toleriert!

Studienbegleitende Prüfung

- nicht alle SoS-Studiengänge sind den gleichen Randbedingungen unterworfen:
 - der Schein ist $\begin{cases} \textit{Empfehlung} & \text{für WIF/Mathematik-Studierende} \\ \textit{Pflicht} & \text{für alle anderen} \end{cases}$
- die Prüfung erfolgt schriftlich: Klausur (120 Minuten), Zeitraum 09/10
 - geprüft wird Stoff bzw. Inhalt der Vorlesung *und* der Übung !
- bei aktiver Mitarbeit in den Übungen ist die Klausur „leicht“ zu bestehen
 - die Programme im Team entwickeln, aber selbst zum Laufen bringen

Kontakt

..... http://www4.informatik.uni-erlangen.de/~*

Dozenten

- Jürgen Kleinöder [~jklein](#)
- Wolfgang Schröder-Preikschat [~wosch](#)



Mitarbeiter

- Stefan Steidl ³
- Christian Wawersich [~wawi](#)



³<http://www5.informatik.uni-erlangen.de/~steidl>

Fragen?