

Vorlesung Informatik III (Systemsoftware) — Informationen —

Daten der Vorlesung

Titel:	Informatik III (Systemsoftware)
Termin:	Montags, 10 ¹⁵ – 11 ⁴⁵ , und Donnerstags, 12 ⁰⁰ – 13 ³⁰
Beginn:	15. Oktober 2001
Ort:	Hörsaal des Mathematischen Instituts, Arndtstr. 2, Eingang Fichtestr., 1. Stock
Betreute Übung:	Dienstags, 12 ⁰⁰ – 13 ³⁰
Beginn:	23. Oktober 2001
Ort:	Raum 226, Hochschulrechenzentrum, Heinrich-Buff-Ring 44
WWW Seite:	http://www.informatik.uni-giessen.de/staff/brass/iii01/
Dozent:	Prof. Dr. Stefan Brass
Telefon:	0641/99-32150 (Büro), 0641/9609992 (privat, nur 8 ⁰⁰ – 21 ³⁰)
Email:	Stefan.Brass@informatik.uni-giessen.de
Büro:	Raum 108, Arndtstr. 2
Sprechstunde:	Donnerstags, 15 ⁰⁰ – 16 ⁰⁰
Tutor:	Herr Dierk Michel
Email:	Dierk.Michel@math.uni-giessen.de
Sekretariat:	Frau Danuta Jäger, Raum 120, Arndtstr. 2 (nur vormittags)
Telefon:	0641/99-32141
Telefax:	0641/99-32149

Ziele der Vorlesung

Diese Vorlesung ist Teil eines viersemestrigen Zyklus, der die Grundlagen der Informatik vermitteln soll.

- In Informatik I wird in die Programmierung eingeführt anhand der Sprache Pascal. Diese Vorlesung gibt außerdem eine Einführung in die Informatik und lehrt einige Algorithmen und Datenstrukturen.

- Informatik II ist eine Einführung in die theoretische Informatik. Dort werden Formalismen zur Syntaxdefinition vorgestellt (insbesondere reguläre Ausdrücke und kontextfreie Grammatiken), außerdem werden Berechenbarkeit und Grundlagen der Komplexitätstheorie behandelt.
- Informatik IV, das „Informatik Praktikum“ behandelt Netzwerke und UNIX (aus Benutzersicht: Befehle, Shellprogrammierung, etc.).

Ziel dieser Vorlesung (Informatik III) ist es nun, einen Blick unter die Oberfläche eines Rechners zu werfen, und wenigstens ansatzweise zu verstehen, wie ein Rechnersystem funktioniert. Dabei soll die Hardware nur relativ kurz behandelt werden, etwas ausführlicher dagegen zwei wichtige Bausteine der Systemsoftware:

- Das Betriebssystem, das eine angenehmere Schnittstelle zur Hardware bietet, und die vorhandenen Betriebsmittel (z.B. Prozessor, Speicher, Platte, Drucker) zwischen mehreren Benutzern verteilt.
- Ein Compiler, der ebenfalls eine angenehmere Schnittstelle zur Hardware bietet, indem er eine Programmierer-freundliche Hochsprache in die Maschinensprache des Rechners übersetzt.

Andere wichtige Systemsoftware, insbesondere Shells (Kommandointerpreter) und Netzwerkdienste wie Mail werden im Informatik Praktikum behandelt.

Kenntnisse von Betriebssystem von Compiler werden Ihnen z.B. in der System-nahen Programmierung helfen. Allgemein ist es gut, wenn etwas nicht funktioniert, sich in den Computer „hineinversetzen zu können“, d.h. eine ungefähre Vorstellung davon zu haben, wie diese Dinge intern funktionieren. Auf diese Art können Sie vielleicht andernfalls eher kryptische Fehlermeldungen verstehen. Auch können Sie eventuell die Leistung (Performance) Ihrer Programme oder des Betriebssystems mit entsprechenden Kenntnissen steigern. Schließlich befriedigt es auch die Neugier, wenigstens ein bißchen zu verstehen, wie die Werkzeuge funktionieren, die man täglich benutzt.

Sowohl über Betriebssysteme wie über Compiler werden Spezialvorlesungen angeboten. In Informatik III geht es nicht um experimentelle Entwicklungen und neue Forschungsergebnisse, sondern um das, was noch zur informatischen Allgemeinbildung gerechnet werden kann.

Um konkret werden zu können, und sowohl die Verwendung von Betriebssystem-Aufrufen und Eingaben für den Compiler zu zeigen, als auch einige Algorithmen und Datenstrukturen aus Betriebssystem und Compiler vorführen zu können, muß in dieser Vorlesung natürlich Programmcode gezeigt werden. Die Sprache Pascal ist für die systemnahe Programmierung aber nicht sehr geeignet. In dieser Vorlesung wird daher in die Programmiersprache C eingeführt, die dann auch für Programmbeispiele verwendet wird. Die Umstellung von Pascal auf C sollte nicht sehr schwierig sein, und im Laufe Ihres Lebens werden Sie sich noch in viele Programmiersprachen einarbeiten müssen. Der Dozent dieser Vorlesung ist 37 Jahre alt und kennt/kannte ungefähr 20 Sprachen (Algol 60, Algol 68, Basic, Z80 Maschinensprache, C, IBM 370 Assembler, Fortran, 68000 Assembler,

Lisp, Prolog, TeX, BibTeX, Pascal, Modula 2, Datalog, SQL, PL/SQL, etwas Postscript, HTML, XML, C++).

Die Fähigkeit, in C programmieren zu können, inklusive systemnaher Programmierung, ist dann auch ein wichtiges Ziel dieser Vorlesung, vielleicht das wichtigste Ziel. Zumindest läßt sich diese Fähigkeit leicht in Hausaufgaben und Klausuren abprüfen, und kann später im Beruf auch ganz offensichtlich eingesetzt werden. Ich wäre sehr enttäuscht, wenn ein Student/eine Studentin, der/die vier Semester an Lehrveranstaltungen der Informatik teilgenommen hat, hinterher die wichtige handwerkliche Grundlage der Programmierung nicht beherrscht. Andere Informatik-Inhalte können dann nicht wirklich konkret werden.

Ein Thema, das in dieser Vorlesung auch immer wieder vorkommen wird, sind Datentypen/Datenstrukturen und Algorithmen. Dieses Thema soll etwas systematischer bereits in Informatik I eingeführt worden sein. In Informatik III werden die Datentypen von C definiert und einige wichtige Datenstrukturen betrachtet, die in Compilern und Betriebssystemen eingesetzt werden.

Am Ende des Semesters soll noch etwas auf Windows-Programmierung eingegangen werden, um einen Eindruck von der Erstellung graphischer Benutzeroberflächen zu geben. Die Übungen finden auch an Windows-Rechnern statt. UNIX ist erst Gegenstand des Informatik-Praktikums im vierten Semester. Da UNIX ein elegantes und klassisches Betriebssystem ist, werden seine wichtigsten Systemaufrufe und einige Interna aber im Abschnitt über Betriebssysteme behandelt. MS-DOS war dagegen ein schlechter Entwurf, und das darauf aufbauende Windows leidet sehr unter der historischen Last. Windows NT/2000 ist ein besserer Neuentwurf, der viel von UNIX gelernt hat.

Wenn es die Zeit zuläßt, wird es auch eine kurze Einführung in Datenbanksysteme geben (das Spezialgebiet des Dozenten). Datenbanksysteme haben zum Teil ganz ähnliche Funktionen wie Betriebssysteme, und es spricht viel dafür, daß in der Zukunft Betriebssysteme und Datenbanksysteme verschmolzen werden.

Geplante Themen

- Die Programmiersprache C
- Einführung in Computer-Hardware (kurz)
- Einführung in Compiler
- Einführung in Betriebssysteme
- Einige Datenstrukturen (für Compiler und Betriebssystem)
- Einführung in Datenbanksysteme (kurz, entfällt eventuell)
- Einführung in Grafik-Programmierung, Benutzerschnittstellen

Ich halte diese Vorlesung zum ersten Mal. Gewisse Abweichungen von der Planung sind daher leicht möglich.

Voraussetzungen zur Teilnahme

Normalerweise sollten die Teilnehmer dieser Vorlesung vorher bereits erfolgreich an Informatik I und II teilgenommen haben. Dies kann aber durch entsprechende Vorkenntnisse ersetzt werden. Folgende Vorkenntnisse sind notwendig:

- Sie sollten eine Programmiersprache beherrschen. Dies kann z.B. Pascal sein, aber auch eine andere imperative Programmiersprache. Wenn Sie C oder C++ bereits kennen, könnten Teile der Vorlesung eventuell etwas langweilig für Sie werden. Es gibt aber noch genug anderes Material, und eine systematische Wiederholung ist eventuell nützlich.
- Sie sollten reguläre Ausdrücke und vor allem kontextfreie Grammatiken kennen.
- Sie sollten die booleschen Werte „wahr“ und „falsch“ sowie die Verknüpfungen „und“ und „oder“ kennen. Allgemeine mathematische Grundlagen wie Mengen und Funktionen sind natürlich auch notwendig.
- In Informatik I wurden auch Datenstrukturen wie Stacks und Hashtabellen besprochen. Sollten diese Datenstrukturen nicht allgemein bekannt sein, werden sie wiederholt.
- Englische Sprachkenntnisse sind notwendig. Ich werde in Deutsch sprechen, aber meine Folien sind in Englisch.

Wichtig ist auch, daß Sie genug Zeit mitbringen, um ein Lehrbuch zu lesen, sich mit den Hausaufgaben zu beschäftigen, und relativ viel praktisch zu programmieren. Wieviel Sie in diesem Kurs lernen, hängt ganz wesentlich davon ab, wieviel Zeit Sie investieren.

Kurz-Information über den Dozenten

PERSÖNLICHE DATEN

Name: Stefan Brass
Jahrgang: 1964
Familienstand: Verheiratet mit Nina Brass, Tochter Elisabeth (* 13.9.2000)
Sprachen: Deutsch, Englisch, Russisch

AUSBILDUNG

Diplom: TU Braunschweig, 1988 (Studium der Informatik von 1982 bis 1988).
Promotion: Universität Hannover, 1992.
Habilitation: Universität Hannover, 1997.

BERUFSTÄTIGKEIT

1988–1990	Wissenschaftlicher Mitarbeiter an der Universität Dortmund.
1990–1992	Wissenschaftlicher Mitarbeiter an der Universität Hannover.
1992–1996	Wissenschaftlicher Assistent an der Universität Hannover.
1996–1998	Verwalter der vakanten Professur für Informationssysteme an der Universität Hildesheim.
1998–2000	„Assistant Professor“ im „Department of Information Science and Telecommunications“ an der University of Pittsburgh.
Seit 7/2000	Professor (C3) für Informatik, Schwerpunkt Datenbanken, an der Justus-Liebig-Universität Gießen.

FORSCHUNGSGEBIETE

Deduktive Datenbanken, DB Ausbildung, XML Anfragesprachen, XML Datenbanken, WWW Suchmaschinen, Logische Programmierung, Wissensrepräsentation.

Lehrbücher

Aufgrund der Vielzahl behandelter Themen gibt es kein Lehrbuch, das die Inhalte dieser Vorlesung komplett abdeckt. Dafür gehen die Lehrbücher häufig deutlich mehr ins Detail als es in dieser Vorlesung möglich ist. Wenn Sie sich nur ein Buch anschaffen können, und Sie schon eine Programmiersprache kennen, aber nicht C, dann würde ich das Lehrbuch der C-Erfinder Kernighan und Ritchie empfehlen. Dieses Buch ist auch in der Universitäts-Bibliothek in mehreren Exemplaren vorhanden. Können Sie sich noch ein zweites Buch beschaffen, scheint mir das Buch von Tanenbaum, „Modern Operating Systems“, am interessantesten. Ist noch ein drittes Buch möglich, wäre z.B. das Buch von Wirth über Compilerbau eine kompakte Einführung in dieses Thema. Wenn Sie sich im Compilerbau vertiefen möchten, sollten Sie aber eines der anderen Bücher wählen (Aho/Sethi/Ullman ist das klassische Werk, Fischer/LeBlanc etwas praktischer, Güting/Erwig ein gutes modernes deutsches Lehrbuch). Wenn Sie sich in die Programmierung graphischer Oberflächen unter Windows in C einarbeiten wollen, scheint mir das Buch von Petzold gut.

C/C++:

- Kernighan, Brian W. / Ritchie, Dennis M.: C Programming Language, 2nd Ed. Prentice Hall, ISBN 0-13-110370-9, 272 Seiten, ca. DM 137,76.
- Kernighan, Brian W. / Ritchie, Dennis M.: Programmieren in C. ANSI C (2. A.). Mit dem C- Reference Manual. Hanser, 1990, ISBN -446-15497-3, DM 65,80.
- Stroustrup, Bjarne: The C++ Programming Language, Special Edition. Addison Wesley, 2000, ISBN 0201700735, 1040 Seiten, ca. DM 151,81.
- Stroustrup, Bjarne: Die C++-Programmiersprache, 4 Auflage. Deutsche Übersetzung der Special Edition.

Addison-Wesley/Pearson, 2000, ISBN 382731660X, 1108 Seiten, DM 99,90.

Hardware, Computerarchitektur:

- Messmer, Hans-Peter: PC - Hardwarebuch. 6.Aufl.
Addison-Wesley, 2000, ISBN 3827314615, DM 119,90.
- Tanenbaum, Andrew S. / Goodman, G.:
Structured Computer Organization, 4th Edition.
Pearson Higher Education, 1998, ISBN 0-13-095990-1, 669 Seiten, ca. DM 133.30.
- Tanenbaum, Andrew S. / Goodman, James:
Computerarchitektur: Strukturen. Konzepte. Grundlagen. 4. Auflage.
Addison-Wesley/Pearson Studium, 2001, ISBN 3-8273-7016-7, 772 Seiten, DM 89.95.

Betriebssysteme:

- Tanenbaum, Andrew S.: Modern Operating Systems.
Prentice Hall/Pearson Higher Ed., 2001, ISBN 0-13-031358-0, 752 S., ca. DM 98.99.
- Brause, Rüdiger: Betriebssysteme. Grundlagen und Konzepte.
Springer-Verlag, 2001, ISBN 3540675981, 390 Seiten, DM 55.90.
- Solomon, David A. / Russinovich, Mark:
Inside Microsoft Windows 2000, English, 3rd Edition, w. CD-ROM.
Microsoft Press Corp., 2000, ISBN 0-7356-1021-5, 903 Seiten, ca. DM 107.00.
- Solomon, David A. / Russinovich, Mark:
Inside Microsoft Windows 2000, Deutsch, 3. Auflage, m. CD-ROM.
Microsoft Press, 2000, ISBN 3-86063-630-8, DM 129,01.
- Petzold, Charles: Programming Windows, 5th Edition.
Microsoft Press International, 1998, ISBN 1-57231-995-X, 1479 S., ca. DM 145,20.
- Petzold, Charles:
Windows Programmierung. Das Entwicklerhandbuch zur WIN32-API. 5. Auflage.
Microsoft Press, 1999, ISBN 3860634879, 1350 Seiten, DM 118,99.
- Simon, Richard J.: Microsoft Windows 2000 API SuperBible, w. CD-ROM.
Sams, 2000, ISBN 0-672-31933-0, 1500 Seiten, ca. DM 140,95.
(Bei der deutschen Übersetzung fehlen die letzten Kapitel in der gedruckten Form, sie sind nur auf der CD.)
- Richter, Jeffrey: Programming Applications for Windows, 4th Edition.
Microsoft Press International, 1999, ISBN 1572319968, 1056 Seiten, ca. DM 186,45.
- Richter, Jeffrey: Microsoft Windows- Programmierung für Experten, 4. Auflage.
Microsoft Press, 2000, ISBN 3860636154, DM 129,01.

- Bach, Maurice J.: The Design of the UNIX Operating System. Prentice Hall/Pearson Higher Education, 1986, ISBN 0132017997 (hardback), ISBN 0-13-201757-1 (paperback), 903 Seiten, ca. DM 165,01.
- Marshall Kirk McKusick, Keith Bostic, Michael J. Karels, John S. Quarterman: The Design and Implementation of the 4.4 BSD Operating System (Reissued 2nd Ed). Addison Wesley, 1996, ISBN 0201549794, 550 Seiten, ca. DM 145,20.

Compilerbau:

- Wirth, Niklaus: Grundlagen und Techniken des Compilerbaus. Oldenbourg, 1995, ISBN 3486243748, 195 Seiten, DM 38,73.
- Güting, Ralf H. / Erwig, Martin: Übersetzerbau. Techniken, Werkzeuge, Anwendungen. Springer Verlag, 1999, ISBN 3540653899, 367 Seiten, DM 65,90.
- Aho, Alfred V. / Sethi, Ravi / Ullman, Jeffrey D.: Compilers - Principles, Techniques, and Tools. Addison-Wesley Longman, Repr. 2000, ISBN 0-201-10088-6, 796 S., ca. DM 120,95.
- Aho, Alfred V. / Sethi, Ravi / Ullman, Jeffrey D.: Compilerbau, Teil 1. Oldenbourg, 1999, ISBN 3-486-25294-1, DM 68,06.
- Aho, Alfred V. / Sethi, Ravi / Ullman, Jeffrey D.: Compilerbau, Teil 2. Oldenbourg, 1999, ISBN 3-486-25266-6, DM 68,06.
- Fischer, Charles N. / LeBlanc, Richard J.: Crafting a Compiler with C. Pearson Higher Education, 1991, ISBN 0-8053-2166-7, ca. DM 110,24
- Fischer, Charles / Leblanc, Richard / Cytron, Ron: Crafting a Compiler Featuring Java. Addison Wesley / Pearson Higher Ed., 2000, ISBN 0-201-38593-7, ca. DM 102,18.

Anwesenheit in der Vorlesung

Alle ausgeteilten Materialien werden (eventuell mit kleiner Verzögerung) auf der WWW-Seite dieser Vorlesung verfügbar gemacht. Daraus sollten zumindest die behandelten Themen deutlich werden, zum Nachholen des Stoffes könnte allerdings noch ein Lehrbuch oder eine Mitschrift nötig sein. Normalerweise macht das Nachholen mehr Arbeit als die Teilnahme an der Vorlesung. Ich denke auch, daß die Diskussionen, kleinen Aufgaben und Beispiele während der Vorlesung sehr wichtig sind. Es könnte auch wichtige Ankündigungen geben, die Sie verpassen, wenn Sie nicht zur Vorlesung kommen. Es ist also in Ihrem eigenen Interesse, normalerweise zur Vorlesung zu kommen.

Verwendete Software in den Übungen

Für Übungen/Hausaufgaben benötigen Sie einen C-Compiler. Im Raum 226 des HRZ ist Microsoft Visual C++ installiert (C ist eine Teilmenge von C++). Zu der Übungszeit Dienstags 12–14 ist dieser Raum für uns reserviert. Zu dieser Zeit wird dort der Tutor, Herr Michel, und zum Teil auch der Dozent anwesend sein.

Falls Sie keine Betreuung benötigen (oder nur ganz wenig per Email), können Sie die Hausaufgaben auch zu anderen Zeiten und auch zu Hause bearbeiten. Es gibt keine Tafelübung, der betreute Übungstermin dient allein der Bearbeitung der Hausaufgaben.

Es ist wesentlich für Ihren Erfolg in diesem Kurs, daß Sie wirklich praktische Erfahrungen in der Programmierung sammeln.

Wenn Sie die Aufgaben zu Hause bearbeiten wollen, brauchen Sie einen C-Compiler. Microsoft Visual C++ (Standard Edition, Englisch) kostet zur Zeit ca. DM 259 (eventuell gibt es Rabatte für Studierende).

Den C-Compiler der Firma Inprise/Borland ohne die Entwicklungs-Umgebung können Sie kostenlos unter folgender URL bekommen:

- <http://www.inprise.com/bcppbuilder/freecompiler/>

Mit Entwicklungsumgebung kostet der Borland C++ Builder 5 Standard allerdings auch DM 249. Unter der URL

- <http://www.willeboordse.demon.nl/>

soll es eine einfache freie Entwicklungsumgebung geben.

Wenn Sie unter UNIX/Linux arbeiten, ist der kostenlose GNU C/C++ Compiler erstklassig. Eine ältere Portierung auf MS-DOS/Windows ist DJGPP, verfügbar unter

- <http://www.delorie.com/djgpp/>
- <http://www.gnu.org/directory/djgpp.html>

Es läuft allerdings nicht unter Windows 2000 und unterstützt nur DOS-Programmierung. Cygwin ist ein neuerer Versuch, GNU-Tools für Windows zur Verfügung zu stellen. Es ist verfügbar unter

- <http://sources.redhat.com/cygwin/>

Dies würde im wesentlichen eine Bibliothek installieren, die ein Windows-System sehr UNIX-ähnlich macht.

Weitere C-Compiler, die ich im WWW gefunden habe, sind

- <http://www.bloodshed.net/devcpp.html>
- <http://www.cs.virginia.edu/lcc-win32/>
- <http://sunset.backbone.olemiss.edu/bobcook/eC/>

Alle diese Links finden Sie auch auf der WWW-Seite der Vorlesung. Ich habe persönlich allerdings nur den GNU C-Compiler, Microsoft Visual C++ und eine alte Version von Borland C ausprobiert (sämtlich sehr gut).

Bitte teilen Sie Ihre Erfahrungen mit den Compilern dem Dozenten mit, andere Studenten können dann davon profitieren. Wenn Sie Ihre Erfahrungen relativ sorgfältig aufschreiben, können Sie auch Extrapunkte dafür bekommen. Selbstverständlich ist der Dozent auch für nützliche Referenzen für die WWW-Seite dankbar (und honoriert das eventuell mit Extrapunkten).

Kriterien für Übungsscheine

Es gibt zu dieser Vorlesung benotete Übungsscheine. Ich werde für jeden Teilnehmer berechnen, wie viele Prozent aller zu vergebenen Punkte er/sie bekommen hat. Die Punkte teilen sich folgendermaßen auf:

- 35% für die Hausaufgaben. Hausaufgaben werden mehr oder weniger jede Woche am Montag ausgegeben, und müssen spätestens in der Übung am Dienstag der darauffolgenden Woche abgegeben werden. Selbstverständlich können Sie auch in der Vorlesung abgegeben werden. Sie können die Hausaufgaben in Gruppen von bis zu drei Studierenden bearbeiten, oder auch einzeln, wenn Sie das vorziehen.
- 25/40% für die Klausur am 10. Dezember. In beiden Klausuren können Sie die Vorlesungsmaterialien, Notizen, und beliebige Bücher benutzen, aber keine Notebooks. Die bessere Klausur wird mit 40% gewichtet, die schlechtere mit 25%.
- 40/25% für die Klausur am 7. Februar.
- Bis zu 5% an Extrapunkten können Sie für besondere Aktivitäten bekommen. Wenn Sie z.B. einen Fehler in den Materialien (Folien, Hausaufgaben) finden, bekommen Sie dafür normalerweise ca. 1% an Pluspunkten (nur für den/die jeweils Ersten, Ausnahmen sind möglich). Falls Sie mir durch häufige Fragen während der Vorlesung oder durch mehrmaliges freiwilliges Vorrechnen positiv auffallen, gibt es dafür eventuell auch Pluspunkte. Sie können mir auch ein paar Seiten mit Verbesserungsvorschlägen für diese Vorlesung, weiteren Beispielen, oder Erfahrungen aus praktischen Anwendungen schreiben, um an Pluspunkte zu kommen.
- Alternativ können auch beide Klausuren mit jeweils 50% gewichtet werden. Falls sich dadurch eine bessere Zensur ergibt, werde ich diese Zensur verwenden. Die Hausaufgaben werden dann nicht berücksichtigt. Sie sind aber eine wichtige Vorbereitung für die Klausuren.

Der Prozentsatz von der Maximalpunktzahl wird dann gemäß folgender Tabelle in eine Zensur umgerechnet:

%	≥ 90	87–89	83–86	80–82	77–79	73–76	70–72	67–69	63–66	60–62
Note	1.0	1.3	1.7	2.0	2.3	2.7	3.0	3.3	3.7	4.0

Es ist möglich, daß ich die Grenzen etwas nach unten verschiebe, oder die Regeln in anderer Weise leicht verändere, so daß sich die Zensuren verbessern. Eventuell macht auch Ihre Prüfungsordnung Änderungen nötig.

Hausaufgaben, Gruppenarbeit

Es ist sehr wichtig, daß Sie sich selbst mit der praktischen Anwendung des Stoffes in Hausaufgaben auseinandersetzen. Es reicht nicht aus, nur der Vorlesung zuzuhören.

Die Hausaufgaben sind auch eine wichtige Vorbereitung für die Klausuren. Zumindest ein Teil der Aufgaben in den Klausuren wird ähnlich zu früheren Hausaufgaben sein. Programmierung muß man praktisch üben. Die Hausaufgaben geben Ihnen und mir auch wichtige Rückmeldungen.

Die Hausaufgaben sind nicht sehr umfangreich und können einzeln bearbeitet werden. Manche Studierende haben aber sehr gute Erfahrungen mit Gruppenarbeit gemacht. Sie können daher die Hausaufgaben auch in Gruppen von bis zu drei Personen abgeben. Denken Sie aber nicht, daß Sie dadurch Arbeit sparen! In erfolgreichen Gruppen kommt jedes Mitglied bereits mit einem Entwurf der Lösung zum Gruppentreffen, so daß dann nur über die Vor- und Nachteile der verschiedenen Lösungen diskutiert werden muß. Wenn Sie die anderen Gruppenmitglieder die Arbeit machen lassen, werden Sie in den Klausuren eine unangenehme Überraschung erleben. Ihr Lernerfolg hängt ganz wesentlich von der eingesetzten Arbeit ab.

Wenn Sie wollen, können Sie Gruppen beliebig wechseln. Die Punkte für eine Lösung werden jedem Gruppenmitglied individuell gutgeschrieben.

Abgabefristen für Hausaufgaben

Hausaufgaben werden jeweils am Montag ausgegeben und müssen spätestens am Dienstag der nächsten Woche in der Übung abgegeben werden. Ich akzeptiere Abgaben in Ausnahmefällen auch noch am Donnerstag bevor die Vorlesung beginnt. Falls Sie Hausaufgaben später abgeben, werden Ihnen Punkte abgezogen (normalerweise ca. 20%). Bei mehr als einer Woche Verspätung oder mehr als dreimaliger Verspätung wird Ihre Abgabe eventuell nicht mehr korrigiert.

Achten Sie besonders vor den Klausuren darauf, daß Sie Ihre Hausaufgaben rechtzeitig abgeben. Sonst bekommen Sie die korrigierte Version eventuell nicht mehr vor der Klausur zurück.

Korrektur von Hausaufgaben und Klausuren

Es ist sehr wichtig, daß Sie verstehen, warum Sie Punkte verloren haben. Einerseits lernt man aus Fehlern, und andererseits machen Tutoren und Professoren auch Fehler. Bitte fragen Sie, wenn Ihnen unklar ist, warum eine Lösung falsch ist.

Falls keine Punkte abgezogen wurden, und Ihnen die Lösung aber nachträglich falsch vorkommt, fragen Sie bitte auch. Ich werde in diesem Fall keine Punkte mehr abziehen.

Wenn Sie allerdings für eine Teilaufgabe Punkte zurück haben wollen, behalte ich das Recht vor, auch die anderen Aufgaben des Übungsblattes neu zu korrigieren.

Klausuren

Sie können in den Klausuren Ihre Notizen, die ausgeteilten Materialien und beliebige Bücher benutzen. Da nicht alle Studierenden ein Notebook haben, kann ich die Benutzung von Computern nicht erlauben. Sie benötigen auch keinen Taschenrechner für die Klausuren.

Das Ziel dieses Kurses ist nicht, etwas nur auswendig zu lernen. Vielmehr sollen Sie den Stoff verstehen und praktische Fähigkeiten gewinnen.

Beachten Sie bitte, daß die Zeit bei der Klausur beschränkt ist (90 Minuten). Ich habe nicht vor, einen großen Zeitdruck zu erzeugen, aber wenn Sie längere Passagen der Materialien lesen müssen, wird die Zeit nicht ausreichen. Bücher werden Ihnen in der Klausur nur nützlich sein, wenn Sie sie gelesen haben, und daher wissen, wo sie suchen müssen.

Bedenken Sie auch, daß es in der Klausur keine Fehlermeldungen des Compilers mehr gibt. Wenn Sie bei den Hausaufgaben immer sehr lange probieren müssen, bevor Sie die korrekte Formulierung in C finden, werden Sie in der Klausur ein Problem haben: Dort müssen Sie es beim ersten Versuch richtig hinbekommen. Es ist leider nicht möglich, die Klausur in einem Raum mit Rechnern zu veranstalten. Außerdem könnte ein späterer Arbeitgeber auch erwarten, daß Sie einfache Programme Anfragen relativ zügig und sicher schreiben können. Aus diesen Gründen sollten Sie vor den Klausuren ausreichend üben, so daß Sie typische Fehler schon kennen.

Abschreiben, Täuschungsversuche

Sie haben die Möglichkeit, Hausaufgaben in Gruppen zu bearbeiten. Falls Sie unbedingt eine Gruppe von mehr als drei Personen bilden wollen, können wir darüber sprechen.

Aber schreiben Sie nicht ab und geben Sie Ihre Lösung auch niemandem zum Abschreiben. Da Herr Michel alle Lösungen korrigieren wird, ist es recht wahrscheinlich, daß er bemerkt, wenn zwei Lösungen zu ähnlich sind. Der Dozent wird es dann noch einmal überprüfen, und verfügt über einige Erfahrung mit solchen Fällen.

Teils wurden sogar ganz merkwürdige Syntaxfehler abgeschrieben. Wenn so ohne jedes Verständnis abgeschrieben wird, gibt es absolut keinen Lernerfolg durch die Hausaufgaben. Setzen Sie Ihre Zeit besser sinnvoller ein. Wenn Ihre Klausuren gut sind, brauchen Sie die Punkte für die Hausaufgaben nicht. Und die Hausaufgaben helfen Ihnen für die Klausur nur, wenn Sie sie selbst gelöst haben.

Manchmal wird es als sozial angesehen, jemand die Hausaufgaben zum Abschreiben zu geben. Aber das stimmt nicht: Sie nehmen ihm/ihr die Möglichkeit, durch die eigene Beschäftigung mit den Hausaufgaben etwas zu lernen. Um jemandem zu helfen, sollten Sie sich erklären lassen, wo das Problem liegt, und dann Hinweise zur Lösung dieses Problems geben. Es ist außerdem auch möglich, den Tutor oder den Dozenten zu fragen (z.B. per Email). Fragen und Antworten zu den Hausaufgaben, die von allgemeinem Interesse sind, werden auf der WWW-Seite veröffentlicht.

Sollten zwei Hausaufgaben zu ähnlich sein, werden beide Gruppen von der Korrektur weiterer Hausaufgaben ausgeschlossen. Es spielt dabei keine Rolle, wer abgeschrieben hat. Sie können den Schein für diese Vorlesung immer noch bekommen, aber nur über Ihre Leistungen in den Klausuren.

Täuschungsversuche in den Klausuren sind sehr ernst. Ich werde in diesem Fall den Schein für diese Vorlesung verweigern. Bitte beachten Sie, daß es wichtig ist, nicht den Eindruck zu erwecken, Sie hätten versucht, zu schummeln. Ich habe ja immer nur einen Eindruck und keinen Beweis. Bitte tragen Sie Ihren Teil zu einem guten und vertrauensvollen Verhältnis in dieser Vorlesung bei.

Punkte-Datenbank im WWW

Es ist möglich, daß Tutor oder Dozent beim Notieren der Punkte für Hausaufgaben und Klausuren Fehler machen. Aus diesem Grund wurde die Möglichkeit vorgesehen, daß die Kursteilnehmer ihre Punkte im WWW selbst kontrollieren können. Dadurch können Sie jederzeit Ihren aktuellen Stand einsehen und bekommen auch einige Statistiken über die Leistung der anderen Teilnehmer. Die Adresse der Web-Datenbank ist

`http://www.informatik.uni-giessen.de/staff/brass/iii01/view.html`

Sie ist auch ein Beispiel für Web-Datenbank-Schnittstellen, das wir in dieser Vorlesung besprechen werden.

Ihre Daten sind über ein Passwort geschützt, das Sie selbst vergeben können. Damit dies funktioniert, müssen Sie sich in der Datenbank registrieren (d.h. Namen, Vornamen, und Passwort eingeben), bevor die ersten Punkte eingetragen werden. Dies geschieht unter

`http://www.informatik.uni-giessen.de/staff/brass/iii01/reg.html`

Ich kann leider nicht kontrollieren, daß nicht jemand anders Ihren Namen eingegeben hat. Dies würde aber bemerkt, wenn Sie selbst sich registrieren, da es zu jedem Namen/Vornamen nur einen Eintrag in der Datenbank geben kann. Sie können mich dann

informieren. Beachten Sie, daß selbst wenn Sie diese Datenbank nicht benutzen wollen, Sie sich wenigstens registrieren müssen, um Ihre Daten vor anderen zu schützen.

Web Datenbanken sind immer etwas unsicher. Es kann keine Garantie geben, daß es nicht jemanden gelingt, in diese Datenbank einzubrechen. Falls Sie große Bedenken gegen eine Abspeicherung Ihrer Ergebnisse in dieser Datenbank haben, können Sie mir das mitteilen. Dies würde allerdings mir mehr Arbeit machen und Ihnen die Möglichkeit nehmen, Ihren Stand zu kontrollieren.

Bedenken Sie auch, daß Ihr Passwort unverschlüsselt über das Netz geschickt wird (in der Datenbank wird es aber verschlüsselt abgelegt). Sie sollten also ein Passwort verwenden, das Sie nicht auch an anderen wichtigeren Stellen verwenden.

Bitte kontrollieren Sie Ihre Punkte regelmäßig, besonders vor Ausstellung der Scheine am Semesterende.

Fragen in der Vorlesung

Ich freue mich sehr über Fragen während der Vorlesung. Dies zeugt von Ihrem Interesse, und gibt mir die Möglichkeit, bestimmte Punkte noch weiter auszuführen. Ohne Fragen könnte ich ja auch durch ein Buch oder eine Video-Aufzeichnung ersetzt werden. Es ist für mich auch interessanter, Fragen zu beantworten, als mein vorbereitetes Material durchzugehen.

Sie dürfen auch fragen, warum etwas wichtig ist. Ich habe mich nie durch eine Frage eines Studierenden angegriffen gefühlt, und ich denke, daß sich nie ein Studierender durch meine Antwort angegriffen oder bloßgestellt gefühlt hat.

Falls Sie bemerken, daß ich einen Fehler mache, teilen Sie mir das bitte mit. Ich weiß, daß ich von Zeit zu Zeit Fehler mache. Es ist viel besser für alle Teilnehmer, wenn solche Probleme sofort geklärt werden. Falls Sie einen Fehler auf meinen Folien entdecken, gibt es dafür Pluspunkte (bitte schicken Sie mir hinterher eine Email, damit ich das nicht vergesse).

Falls Sie während der Vorlesung nicht fragen mögen, können Sie mir auch Fragen per Email schicken. Wenn es nicht sehr dringend ist, werde ich die gesammelten Fragen am Beginn der nächsten Vorlesungsstunde besprechen. Sollte ich dies vergessen, erinnern Sie mich bitte.

Email Adresse

Falls Sie Fragen zu den Hausaufgaben haben, können Herrn Michel oder mir eine Email schicken:

- Dierk.Michel@math.uni-giessen.de
- Stefan.Brass@informatik.uni-giessen.de

Fragen und Antworten von allgemeinem Interesse sowie Hinweise und Korrekturen werden auf der WWW-Seite zur Vorlesung veröffentlicht (ganz oben). Bitte schauen Sie möglichst zuerst dort nach.

Da ich meine Email auf einem UNIX-Rechner lese, ist das Microsoft Word Format schwierig auszudrucken. Reines ASCII ist am einfachsten für mich, gefolgt von HTML, Postscript, und PDF. Noch andere Formate kann ich im wesentlichen nicht lesen.

Hausaufgaben sollten Sie bitte nur in Ausnahmefällen per Email einreichen, und dann unbedingt an Herrn Michel schicken.

Sprechstunden

Meine Sprechstunde ist Donnerstags, 15⁰⁰ – 16⁰⁰, und nach Vereinbarung.

Wenn Sie eine wirklich kurze Frage haben, können Sie auch zu anderen Zeiten versuchen, ob ich da bin (Mittwochs und Freitags arbeite ich häufig zu Hause). Direkt vor den Vorlesungen (Montag 10, Dienstag 16, Donnerstag 12) habe ich normalerweise keine Zeit.

Verbesserung der Lehre

Gute Lehre ist für mich wichtig. Drei meiner Vorlesungen wurden bei den Umfragen in Pittsburgh bei jeder Frage des offiziellen Fragenkatalogs in das beste Viertel aller Kurse der „School of Information Sciences“ eingeordnet. Darüber habe ich mich natürlich sehr gefreut, und es ist mir ein Ansporn, weiter an der Verbesserung meiner Vorlesungen zu arbeiten. Daher liegt mir sehr daran, ihre Meinung und eventuelle Verbesserungsvorschläge zu erfahren:

- Haben Sie einen Fehler in den Materialien gefunden? Auch einfache Tippfehler oder Fehler der englischen Sprache sind für mich wichtig.
- Welche Abschnitte sind schwierig zu verstehen?
- Was fehlt in der Vorlesung? Dies kann ein ganzer Abschnitt sein, oder auch nur ein kleiner Hinweis, eine Warnung oder Erklärung.
- Haben Sie vielleicht praktische Erfahrungen, die Sie mir mitteilen können?
- Kennen Sie sich mit anderen Betriebssystemen oder Compilern aus und können mir die Unterschiede erklären.
- Haben Sie Vorschläge, wie ich meinen Vorlesungsstil verbessern könnte?
- Kenn Sie WWW-Seiten, die ich in meine Sammlung zu diesem Kurs aufnehmen sollte?

Vorlesungsmaterialien im WWW

Alle Vorlesungsmaterialien werden im WWW unter folgender Adresse verfügbar gemacht:

`http://www.informatik.uni-giessen.de/staff/brass/iii00/`.

Die Folien werden eventuell nur im Postscript-Format zur Verfügung stehen (an der Umsetzung nach PDF arbeite ich noch). Sie können Aladdin Ghostscript zum Ausdrucken verwenden:

`http://www.cs.wisc.edu/~ghost/`