

Informatik I für D-ITET**WS 04/05**URL: <http://www.ti.inf.ethz.ch/ew/courses/inf1-ITET/>

Vorlesungstermin: Mittwoch 8.15–10.00 Uhr. Ort: ETF C1.
Dozent: Joachim Giesen <giesen@inf.ethz.ch>, IFW B48.3.
Chefassistent: Bernhard Seybold <bernhard.seybold@inf.ethz.ch>, IFW C48.1.
Sekretariat: Franziska Hefti <hefti@inf.ethz.ch>, IFW B49.1, Tel. 632 73 92.

Übungstermine:

GRUPPE	TERMIN	RAUM	ÜBUNGSLEITER	EMAIL
A	Mo 14–16	IFW A34	Franziska Meyer	<zismeyer@student.ethz.ch>
B	Mo 14–16	ETZ F91	Cyril Flaig	<cflaig@student.ethz.ch>
C	Mo 14–16	ETZ J91	Ivan Guajana	<guajanai@student.ethz.ch>
D	Mo 14–16	ETZ F6	Simon Heinzle	<sheinzle@student.ethz.ch>
E	Mo 14–16	RZ F21	Paul Drielsma	<paul.drielsma@inf.ethz.ch>
F	Mo 17–19	ETZ E9	Diana Senn	<diana.senn@inf.ethz.ch>
G	Mo 17–19	ETZ F91	Dieter Mitsche	<dmitsche@inf.ethz.ch>
H	Mo 17–19	ETZ G91	Andreas Ottiger	<aottiger@student.ethz.ch>
I	Mo 17–19	ETZ H91	Jürgen Doser	<doserj@inf.ethz.ch>

Achtung: Die ersten Übungen sind am Mo, 25.10.2004. Die erste Übungsserie ist bis zum 1.11.2004, 14:00 Uhr, abzugeben.

Für die Programmieraufgaben stehen die Rechner im ETZ D96 (38 Sun-Workstations), ETL E11 (40 Sun-Workstations), ETL F11 (25 Linux-Workstations), ETZ C99 (14 Sun-Workstations), HG D13 (13 Linux-Workstations), HG E29 (34 Sun-Workstations) sowie HG E26.1 (36 Sun-Workstations) zur Verfügung. Wo vorhanden, können natürlich auch eigene Rechner benutzt werden. Wer mit dem Gedanken spielt, sich ein eigenes Notebook anzuschaffen, der sollte einen Blick auf die Neptun Angebote werfen (<http://www.neptun.ethz.ch/>).

Wir arbeiten mit einer Unix Umgebung. Unter MS Windows kann man eine solche Umgebung sehr einfach mittels *cygwin* emulieren. Auf der Vorlesungswebseite (s.o.) findet man eine Kurzanleitung zur Installation von *cygwin*.

Testatbedingung: Am Ende der Übungen wird ein Testat vergeben. Das Testat erhält, wer

1. sowohl auf den ersten sechs als auch auf den folgenden – wahrscheinlich ebenfalls sechs – Übungsserien mindestens 60% der möglichen Punkte erreicht sowie
2. mindestens die Hälfte der Schnellübungen sinnvoll bearbeitet.

Prüfung: Die Prüfung besteht aus einer Klausur. Bedingung für die Zulassung zur Prüfung ist das Übungstestat. Es sind folgende Hilfsmittel erlaubt: maximal zwei handgeschriebene DIN A4 Blätter (beidseitig beschrieben, also max. vier A4 Seiten).

Abgabe der Übungen: Bearbeitungszeit ist in der Regel eine Woche, bei Programmieraufgaben manchmal auch zwei Wochen. Theoretische Aufgaben werden am Anfang der Übungsstunde abgegeben. Programmieraufgaben werden per Email an den Übungsleiter bis Montag 14.00 Uhr geschickt. Hierbei sind ein paar Punkte zu beachten, die weiter unten aufgelistet sind. Schriftliche Abgaben sind immer mit dem Namen der Vorlesung, der Nummer der Übungsgruppe (A...K), der Nummer des bearbeiteten Übungsblattes und dem eigenen Namen zu beschriften. Genügend Platz zwischen den Aufgaben und ein Rand hilft uns bei der Korrektur. Bitte keinen Bleistift oder roten Stift verwenden und bei Fehlern lieber sauber durchstreichen anstelle von Radieren oder Tipp-Ex.

Schnellübungen: Auf einigen Übungsserien wird die erste Aufgabe aus einer Schnellübung bestehen. Diese Aufgaben sind in den ersten x (typischerweise $x = 20$) Minuten der Übungsstunde zu bearbeiten und werden dann vom Übungsleiter eingesammelt. Es versteht sich, dass hierfür die Anwesenheit in der

Übungsstunde erforderlich ist, eine nachträgliche Abgabe ist nicht möglich. Schnellübungen werden in der Regel eine Woche vorher angekündigt.

Programmieraufgaben: Wie bereits erwähnt, erfolgt die Abgabe der Programmieraufgaben per Email an den Übungsleiter. Um uns die Verwaltungsarbeit zu erleichtern, sollten dabei ein paar Spielregeln beachtet werden.

- Nur die Programmtexte (*sourcecode*) verschicken, nicht die daraus mit Hilfe des Compilers generierten ausführbaren Programme (*executables*).
- Für jede Übungsserie nur eine Mail, und zwar mit *Subject* "Serie <Nummer>", wobei <Nummer> entsprechend durch die Nummer der bearbeiteten Serie ersetzt wird, z.B. "Serie 1" . Sollte doch eine weitere Mail notwendig werden, weil Euch z.B. Fehler aufgefallen sind, so stellt bitte im Subject das Wort "UPDATE:" voran.
- Die verschiedenen Programmtexte werden jeweils als *attachments* vom Typ Text/Plain an die Mail angehängt.
- In Emails und Programmtexten: Nie mehr als 80 Zeichen pro Zeile!
- Programmtexte müssen vom Compiler ohne Fehlermeldungen akzeptiert werden.
- Jeder Programmtext enthält ganz am Anfang in Form eines Kommentars folgende Angaben: Programmname, Autor, Übungsgruppe und bearbeitete Übungsserie sowie Aufgabe. Beispiel:

```
// Informatik - Serie 13 - Aufgabe 4e
// Programm: my_program.C
// Autor:    X. M. Plestudent (Gruppe D)
```

- In den Programmen sind nur die Sprachkonstrukte zu verwenden, die in der Vorlesung oder Übung besprochen wurden.
- Programmtexte sind für Menschen geschrieben, auch wenn sie gleichzeitig von Rechnern verstanden werden müssen. Achtet deshalb auch auf das optische Erscheinungsbild Eurer Programme! Insbesondere zählt hierzu konsequentes und konsistentes Einrücken (*indenting*) und die ebensolche Verwendung von Leerzeichen z.B. vor und nach Operatoren und Schlüsselwörtern (*spacing*). Nehmt Euch die Programme aus der Vorlesung als Beispiel, dann kann eigentlich nicht viel schief gehen. ;-)

Zu guter Letzt: Bei Fragen wendet Euch bitte an Euren Übungsleiter oder bei weitergehenden Problemen an den Chefassistenten Bernhard Seybold. Email ist dafür ein bequemer Weg. Wir werden uns bemühen, diese einmal am Tag zu bearbeiten und zu beantworten, was bei Abwesenheit aber auch einmal länger dauern kann.

Es bleibt uns noch, Euch einen guten Anfang und viel Spass zu wünschen!

Literatur: (Polybuchhandlung, Raum MM B97.1)

Andrew Koenig and Barbara E. Moo: *Accelerated C++*, Addison-Wesley, 2000.

Stanley B. Lippman: *C++ Primer*, 3. Auflage, Addison-Wesley, 1998.

Bjarne Stroustrup: *The C++ Programming Language*, 3. Auflage, Addison-Wesley, 1997.