




# Grundlagen der Programmierung in C

## Einleitung

Wintersemester 2005/2006  
 G. Zachmann  
 Clausthal University, Germany  
[zach@in.tu-clausthal.de](mailto:zach@in.tu-clausthal.de)



## Die Personen



**Gabriel Zachmann**  
*Vorlesung*  
 Institut für Informatik  
 Julius-Albert-Str.4  
 Raum 205  
 E-Mail: [zach@in.tu-clausthal.de](mailto:zach@in.tu-clausthal.de)  
 Tel.: 05323 - 72 71 13  
 Sprechstunde: Mi, 15-16:00



**Carsten Giesemann**  
*Übungsbetrieb*  
 Institut für Informatik  
 Julius-Albert-Str.4  
 Raum 215  
 E-Mail: [giesemann@in.tu-clausthal.de](mailto:giesemann@in.tu-clausthal.de)  
 Tel.: 05323 - 72 71 21  
 Sprechstunde: immer, wenn da

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 2



## Handwerkszeug vs. Kunst

- Erst die perfekte Beherrschung des Handwerkszeugs ermöglicht Kreativität und Kunst
  - Knuth: "The Art of Programming"
  - Zen-Kaligraphen
  - τέχνη (techné = "Kunst" und auch "Handwerk")
- "The society which scorns excellence in plumbing because plumbing is a humble activity, and tolerates shoddiness in philosophy because philosophy is an exalted activity, will have neither good plumbing nor good philosophy. Neither its pipes nor its theories will hold water." (John W Gardner)

Herrigel:  
*Zen in der Kunst des Bogenschießens*

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 3



## Die 3 Stufen des Programmierens

<ul style="list-style-type: none"> <li>▪ Musik:</li> <li>▪ Noten lesen &amp; schreiben</li> <li>▪ Einfache Stücke schreiben, analysieren</li> <li>▪ Bach, Mozart, Strawinsky, ...</li> </ul>	<ul style="list-style-type: none"> <li>▪ Programmieren:</li> <li>▪ Syntax beherrschen</li> <li>▪ Kleinere Programme zustande bringen, Idiome</li> <li>▪ "Wizards", "Gurus", ...</li> </ul>
--	--

Learning by Doing!

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 4



## Ziele der Vorlesung

- *Konzepte* der imperativen und OO Programmierung verstehen
- Diese in der *Praxis* anwenden können
- *Gute* Programme schreiben können
- Definition *gutes Programm* :
  - Gut dokumentiert, modular → leicht zu verstehen & warten
  - Robust, Elegant , Effizient → wenig Bugs & gute Performance
- Ausführlich C/C++

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 5



## Warum nicht eine andere Sprache?

- Java ?
  - Wer C++ kann, lernt auch schnell Java
  - Ist zu einfach ☺
  - Mit C++ lernt man mehr über die Internia
  - "My experience with these folks [auf comp.lang.java] is that most of them can grasp what a pointer is, but less than half ever get 'pointer to pointer'." [Gordon McMillan]
  - "If I am to teach safety I need a system in which I can do dangerous things." [Francis Glasborough]
  - C++ ist effizienter

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 6

**C# ?**

- "It will take a lot to persuade me that the world needs yet another proprietary language (YAPL). It will be especially hard to persuade me that it needs a language that is closely integrated with a specific proprietary operating system. Clearly, I'm no great fan of proprietary languages, and quite a fan of open, formal standards."  
[Bjarne Stroustrup]
- Wer Java und C++ kann, lernt noch schneller C#
- Verantwortung der Uni gegenüber unlauteren Geschäftspraktiken

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 7

**Der wahre Grund ist natürlich ...**

- "C++ is like teenage sex."
  - It is on everyone's mind all the time.
  - Everyone talks about it all the time.
  - Everyone thinks everyone else is doing it.
  - Almost no one is really doing it.
  - The few who are doing it are ...
    - doing it poorly,
    - sure it will be better the next time,
    - not practicing it safely.

(\*) Graffiti found in a toilet stall at the faculty of Computer Science, Technion - IIT, Haifa, Israel, on 8 November 1993.

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 8

**Im Ernst ...**

"C++ was not designed to be a nice, pure language, good for teaching students how to program, ...

... it was designed to be a powerful tool for professional programmers solving real problems in diverse domains. The real world has some rough edges, so it's no surprise there's the occasional scratch marring the finish of the tools on which the pros rely."

(Scott Meyers)

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 9

**Statistiken**

Which programming languages do you currently use or are you planning to move developers to in the next 6 months?

Pittsburgh Regional Delphi Study, March 2003

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 10

**Languages in common use**

Compiled by François Labelle from statistics on open-source projects at SourceForge

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 11

TILOBE Programming Community Index (<http://www.tiobe.com/tpci.htm>) based on the world-wide availability of skilled engineers, courses and third party vendors

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 12

Position	Delta 1 Year	Programming Language	Ratings	Delta 1 Year
1	↑	C	17.122%	-0.65%
2	↓	Java	15.896%	-6.35%
3	==	C++	14.916%	-2.62%
4	↑	(Visual) Basic	11.650%	+4.47%
5	↓	Perl	8.968%	+0.87%
6	==	PHP	8.231%	+2.80%
7	↑↑↑↑	Delphi/Pascal/Kylix	5.865%	+4.28%
8	↑↑↑↑	Python	5.597%	+4.17%
9	↓	SQL	2.693%	-0.55%
10	↓	C#	1.634%	-0.43%

TPC index based on world-wide availability of skilled engineers, courses, and third party vendors, determined by using Google and Yahoo! search engines

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 13

## Interessante Lektüre für Fortgeschrittene

- "The Hundred Year Language":
  - <http://www.paulgraham.com/hundred.html>
  - "Object-oriented programming offers a sustainable way to write spaghetti code."
  - "I can already tell you what's going to happen to all those extra cycles that faster hardware is going to give us in the next hundred years. They're nearly all going to be wasted."

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 14

## Umfrage

- Umfrage zu Ihren Vorkenntnissen / zu Ihrer Motivation:
  - Computer zuhause?
  - Erfahrung mit Betriebssystemen (Windows/Linux)?
  - Internetzugang zuhause?
  - Programmierkenntnisse? (Basic, Pascal, C, C++, Java, sonstige, keine)
  - Schulbildung (Informatik Grundkurs, Informatik Leistungskurs, Mathematik Leistungskurs)
  - Motivation für Studium (Berufsaussichten, SW-Entwicklung, HW-Entwicklung, Projekt-Management, Forschung, Generelles Interesse)
  - Warum Clausthal (Nähe zum Wohnort, Qualität der Lehre, Qualität der Forschung)?

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 15

## Web-Seite zur Vorlesung

- [http://www.in.tu-clausthal.de/~zach/teaching/programming\\_0506/](http://www.in.tu-clausthal.de/~zach/teaching/programming_0506/)
- Vorlesungsfolien
- Übungsblätter
- Online-Literatur und Links
- Downloads

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 16

## Übungsbetrieb

- Voraussetzung zur Klausur: 50% der Punkte!
  - Ca. 13 Übungsblätter, mindestens 25% von jedem Übungsblatt
  - Anmerkung: Die 25% Regel darf pro Gruppe einmal verletzt werden
- Wir machen XP ("Extreme Programming"):
  - 2-er Gruppen (Ausnahmen bitte mit Tutor absprechen)
- Übungsblätter auf der Web-Seite
  - Eine Woche vor dem Abgabetermin der Montagsgruppen
- Abgabe:
  - Jede Woche bei Ihrem Tutor
  - Source & Exec
  - Lauffähig unter Linux auf den PCs im Lab!
  - Jedes Übungsblatt muß vorgeführt werden (ca. 10 min)

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 17

## Bewertungskriterien:

- 50% für korrektes Programm
  - Aufgabenstellung erfüllt
  - Bugfrei
- 50% Stil
  - Sinnvolle Variablen- und Funktionsnamen
  - Kommentare (Parameter einer Funktion, was tun die nächsten 10 Zeilen Code?)
  - Gut lesbarer(!) Code
- Plagiate: Punkte / (Anzahl identischer Codes) für *alle* !

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 18

- Abgabetermin:
  - In der Woche nach Ausgabe zu Ihrem Übungstermin
- Ort der Übungsgruppen:
  - Rechner-Raum im Ifi: Ifi-Pool 2, Raum 302 (Tannenhöhe, rotes Gebäude, Dachgeschoß rechte Seite)
  - Zugang zu allen betreuten Übungszeiten
    - freie Rechner können ab der 3. Vorlesungswoche zum Programmieren benutzt werden
    - Abgaben sind aber nur zur angemeldeten Zeit beim zugewiesenen Tutor möglich
- Anmeldung per eVE ("elektr. Verfahren zur Einschreibung")
  - <http://www.in.tu-clausthal.de/> → eVE (unten links)
  - Account selbst erzeugen ("Neuer User")
  - Ausschuchen eine von "Programmierung I (Uebung 1 ...)"

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 19

- Termine der Übungsgruppen:
  - Alle Zeiten sind s.t. Die Übungszeit beträgt 120 Minuten
  - Mo. 13.00-15.00, 15.00-17.00, 17.00-19.00, 19.00-21.00
  - Di. 8.00-10.00, 10.00-12.00, 15.00-17.00, 17.00-19.00, 19.00-21.00
  - Mi. 8.00-10.00, 13.00-15.00, 15.00-17.00, 17.00-19.00, 19.00-21.00
  - Do. 8.00-10.00, 10.00-12.00, 13.00-15.00, 19.00-21.00
  - Fr. 8.00-10.00, 10.00-12.00, 13.00-15.00, 15.00-17.00

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 20

## Heimarbeit ... (wer hat kein Linux?)

- Irgendeine Distro kaufen und installieren:
  - Z.B.: RedHat, Mandrake, Suse, ...
  - Bouvier?
- Die Knoppix-CD vom RZ ziehen:
  - Keine Installation nötig
  - <ftp://ftp.rz.tu-clausthal.de/pub/linux/knoppix/>
  - ISO ziehen und CD brennen
  - Achtung: Files sichern vor dem Abschalten!
- Cygwin
  - [www.cygwin.com](http://www.cygwin.com)
  - Achtung: Execs laufen nicht auf den Linux-PCs im Pool

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 21

## ... und an der Uni

- Account am Ifi:
  - Jede Gruppe bekommt in der ersten Übung einen generischen Account
  - Die Accounts werden nach dem Semester gelöscht (Daten extern sichern!)
- Remote einloggen auf kaosus
 

```
ssh kaosus.in.tu-clausthal.de -l ckursXXXX
```
- Daten hin- und herkopieren
 

```
scp source-dir ckursXXXX@kaosus.in.tu-clausthal.de:/home/ckursXXXX/...
```

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 22

## Klausur

- Kurz nach dem ersten Semester (evtl. 13.-17.2.)
  - Anmelden über eVE zu gegebener Zeit
- Müssen Nebenfächler die Klausur schreiben?
  - Leider ja: wer einen Schein haben will, muß die Klausur schreiben
- Nebenfächler:
  - BWL: nur die 1/2 VL, letzte Stunde am 8.12.
    - Klausur muß nur bis dahin bearbeitet werden (dürfen natürlich die ganze bearbeiten)
  - Wirtschaftsingenieure: ganze VL!
  - Alle übrigen: dito

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 23

## Literatur

- Bücher zu C/C++:
  - Guido Krüger: *GoTo C Programmierung*. Addison-Wesley.
  - Lippman & LaJoie: *C++ Primer*. Addison-Wesley, März 1998.
  - Bruce Eckels: *Thinking in C++*. Prentice Hall, 2000.
  - Marcus Bäckmann: *Objektorientierte Programmierung für Dummies*. mitp-Verlag, 2002.
- Siehe auch die Online-Literatur auf der VL-Homepage
- Deutsch versus Englisch
  - Englisch ist die *lingua franca* der Informatik!
  - Compiler = Übersetzer = Interpreter?
  - Template = Schablone? Member = Mitglied? Urgh ...

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 24

## Newsgroup und Mailing-Liste

- Newsgroup
  - [tuc.c-kurs](#)
  - Für Fragen aller Art
  - NICHT für Code-Sharing!
  - Funktioniert nur, wenn Kommilitonen Fragen beantworten
  - Tutoren lesen mit (und beantworten hoffentlich auch ab und zu ;-))
- Mailing-Liste
  - [ckurs@in.tu-clausthal.de](mailto:ckurs@in.tu-clausthal.de)
  - E-Mail geht nur an die Tutoren und den Übungsleiter
  - Für organisatorische Fragen, die nicht in die Newsgroup passen

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 25

## Bilder aus der Geschichte

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 26

1978 :

- IBM 3033
- 2 Prozessoren
- 16 MB RAM
- 64 kB Cache
- 8 Mio Dollar (2 Proc)

2003 :

- Apple G4
- 3D Graphik
- 16 GB RAM
- 512 kB L2 Cache
- Ca. 4000 Dollar

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 27

## Bilder heutiger CPU's, Speicher, etc.

Speicher

6502 (1980)

Pentium 4

Mainboard (P4)

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 28

## Von Neumann Architektur

- Daten und Programm *gemeinsam* im Hauptspeicher
- Fundamentaler Instruktionszyklus
- Spezielles Register mit Befehlszähler (*program counter* = PC) enthält Adresse des aktuellen Maschinenbefehls

0 Programm Daten 2<sup>32</sup>-1 Speicher

Eingabegeräte CPU Ausgabegeräte

Central Processing Unit

Arithmetic Logic Unit

Floating Point Unit

ALU FPU Cache Reg (Vorratslager)

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 29

## Fundamentaler Instruktionszyklus

- **Fetch:** Hole den Befehl, dessen Adresse im Befehlszähler steht, aus dem Speicher in das Befehlsregister
- **Increment:** Inkrementiere den Befehlszähler, damit er auf die nächste auszuführende Instruktion weist
- **Decode:** Dekodiere die Instruktion, damit klar wird, was zu tun ist
- **Fetch operands:** Falls nötig, hole die Operanden aus den im Befehl bezeichneten Stellen im Speicher
- **Execute:** Führe die Instruktion aus, ggf. durch die ALU. (Bei einem Sprung wird neuer Wert in das Befehlsregister geschrieben.)
- **Loop:** Gehe zum ersten Schritt (**Fetch**)
- **Analog in High-Level-Sprachen**

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 30

## Was ist Programmieren?

- Kontrolle
  - Computer tut genau, was man gesagt hat (...?)
- Problemlösen
  - Computer tut etwas Sinnvolles
- Kreativ
  - Elegante Lösung finden
- Modellieren
  - Verhalten und Eigenschaften der Teile eines komplexen Systems
- Abstraktion
  - Wichtige Eigenschaften identifizieren und beschreiben, ohne von Details abgelenkt zu werden
- Konkret
  - Abstraktion muß wieder detailliert werden, um Lösung dem Computer "eintrichtern" zu können

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 31

## Was ist ein Programm?

- Griechisch: Reihenfolge, festgelegte Folge, Ziel
- Beispiele:
  - Kochrezept ("Cookbook"-Bücher)
  - Waschmaschinenprogramm (mechanisch)
  - Fernsehprogramm
  - Algorithmus?

```

graph TD
  Problem --> Algo1[Algo 1]
  Problem --> Algo2[Algo 2]
  Problem --> Algo3[Algo ...]
  Algo2 --> Progr1[Progr 1]
  Algo2 --> Progr2[Progr 2]
  
```

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 32

## Historischer Background

1950: Fortran, Lisp

1960: Algol-60, Cobol

1970: Simula, Algol-68, PL/I, Pascal, Prolog, ML

1980: Smalltalk, C, Ada, Miranda

1990: C++, Java

OO lang.      Imperative and Concurrent lang.      Functional lang.

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 33

## Tips zum Studium

- Persönlichen Lernstil entwickeln!
  - Lernen in der Diskussion
  - Gehörtes / Geschriebenes (eigene Handschrift!)
  - Learning by doing
- Fragen Sie!
  - "Wer fragt, ist 5 Minuten lang dumm— wer nicht fragt, bleibt immer dumm."
  - In dieser Vorlesung: mittendrin, im Anschluß, i. d. Sprechstunde

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 34

- Die Mid-Semester-Krise:
  - Man versteht nur noch Bahnhof ...
  - Durchhalten!
  - Psychologische Beratung an der Uni ...
- Man hat genau 1 Leben ...

G. Zachmann Grundlagen der Programmierung in C - WS 05/06 Einleitung, 35