

C/C++ Programmierung

Organisatorisches

Sebastian Hack
Christoph Mallon

`(hack|mallon)@cs.uni-sb.de`

Fachbereich Informatik
Universität des Saarlandes

Wintersemester 2009/2010

Wer sind wir?

Dozent Sebastian Hack
Geb. E1 3 Raum 432
hack@cs.uni-sb.de

Assistent Christoph Mallon
Geb. E1 3 Raum 426
mallon@cs.uni-sb.de

Für uns gilt:

- ▶ Sprechzeiten sind immer, wenn wir da sind
und die Tür offen ist
- ▶ Um sicher zu sein, dass jemand da ist: Vorher schreiben

Voraussetzungen und Ziele

Voraussetzungen:

- ▶ Kenntnisse über imperatives Programmieren (Prog 2)
- ▶ Interesse, etwas über C/C++ zu lernen
- ▶ Spaß am Hacken

Ziele:

- ▶ Überblick über und Einblicke in zwei wichtige Sprachen
- ▶ Sicherer Umgang mit C/C++
- ▶ Kenntnisse über deren Stärken und Tücken

Voraussetzungen und Ziele

Voraussetzungen:

- ▶ Kenntnisse über imperatives Programmieren (Prog 2)
- ▶ Interesse, etwas über C/C++ zu lernen
- ▶ Spaß am Hacken

Ziele:

- ▶ Überblick über und Einblicke in zwei wichtige Sprachen
- ▶ Sicherer Umgang mit C/C++
- ▶ Kenntnisse über deren Stärken und Tücken
- ▶ 3 Punkte

Voraussetzungen und Ziele

Voraussetzungen:

- ▶ Kenntnisse über imperatives Programmieren (Prog 2)
- ▶ Interesse, etwas über C/C++ zu lernen
- ▶ Spaß am Hacken

Ziele:

- ▶ Überblick über und Einblicke in zwei wichtige Sprachen
- ▶ Sicherer Umgang mit C/C++
- ▶ Kenntnisse über deren Stärken und Tücken
- ▶ 3 Punkte
 - Eine Klausur am Semesterende

- ▶ Grundlagen
- ▶ Tücken
- ▶ Programmiertechniken
- ▶ Portabilität
- ▶ Fehlersuche und -vermeidung
- ▶ Programmierstil
- ▶ Projektverwaltung
- ▶ Optimierung

- ▶ „The C Programming Language“ (2nd Ed.) von Brian W. Kernighan und Dennis M. Ritchie
→ Leider lange vor C99
- ▶ „The C++ Programming Language“ (3rd Ed.) von Bjarne Stroustrup
→ Ziemlicher Brocken

Meiden:

- ▶ „C: The Complete Reference“ von Herbert Schildt
- ▶ „C von A bis Z“ von Jürgen Wolf

bullschildt: /bul'shilt/, n.

[comp.lang.c on USENET] A confident, but incorrect, statement about a programming language. This immortalizes a very bad book about C, Herbert Schildt's C - The Complete Reference. One reviewer commented "The naive errors in this book would be embarrassing even in a programming assignment turned in by a computer science college sophomore."¹

¹Aus „The Jargon File (version 4.4.7)“

Im Netz:

- ▶ Usenet C FAQ:
<http://www.faqs.org/faqs/C-faq/faq/>
- ▶ Usenet C++ FAQ:
<http://www.faqs.org/faqs/C++-faq/part1/>

Zum Nachschlagen, wenn man es genau wissen will; nicht zum Lernen geeignet:

- ▶ Entwurf der C99-Norm:
<http://www.open-std.org/JTC1/SC22/wg14/www/docs/n1124.pdf>
- ▶ Entwurf der C++-Norm:
<http://www.open-std.org/JTC1/SC22/wg21/docs/papers/2009/n2914.pdf>


```
float o=0.075,h=1.5,T,r,0,l,I;int L=80,
_,s=3200;main(){for(;s%L||(h-=o,T=-2),
s;4 -(r=0*0)<(l=I*I)|++ _==L&&write(1,(
--s%L?_<L?--_%6:6:7)+"Sascha_\n",1)&&(0
=I=1=_r=0,T+=o /2))0=I*2*0+h,I=1+T-r;}
```

Kritik:

- ▶ kryptisch²
- ▶ veraltet
- ▶ fehleranfällig

²Mehr bei www.ioccc.org

```
class apfel{static float o=0.075f,h=
1.5f,T,r,0,l,I;static int _,s=3200,L
=80;static boolean wrt(int x, char c
, boolean y){ System. out. print(c);
return y;}static void m(boolean b){}
public static void main(String[] args
){for(;s%L!=0||((h==o)<(T=-2))?s!=0:
s!=0;m(4-(r=0*0)<(l=I*I)|++_==L&&wrt
(1,"Sascha_\n" .charAt(--s%L!=0?_<L?
--_%6:6:7),true )&&((0=I=1=r=_=0)+(T
+=o/2)!=0 ))){0=I*2*0+h;I=1+T-r; }}}
```

Kryptische Programme kann man in jeder Sprache schreiben.

Dennis Ritchie:³

C is quirky, flawed, and an enormous success. While accidents of history surely helped, it evidently satisfied a need for a system implementation language efficient enough to displace assembly language, yet sufficiently abstract and fluent to describe algorithms and interactions in a wide variety of environments.

³Aus „The Development of the C Language“

- ▶ Sind weit verbreitet
- ▶ Wird z.B. häufig in der Numerik eingesetzt
- ▶ Die meisten Betriebssysteme sind in C und C++ verfasst
- ▶ Anwendungen wie Bildbearbeitung, Spiele, ...
- ▶ Viele Sprachen erben von C (z.B. Java, C#)
- ▶ C-Übersetzer gibt es für fast jede Plattform
- ▶ Insbesondere C++ bietet Techniken zur Fehlervermeidung
- ▶ Generische Programmierung in C++
- ▶ Wenn es darauf ankommt, lohnt es sich nahe an der Maschine zu sein: Wohl schnellste Hochsprache und ermöglicht dennoch systemnahe Programmierung