

Interaktive Vorlesung

Arvid Terzibaschian

Interaktive Vorlesung

- ▶ **Idee: Wiederholung & Prüfungsvorbereitung**
 - ▶ Vorteil für Studenten: Wissenslücken schließen
 - ▶ Interessant für Dozent, welche Themen besonders schwierig
 - ▶ zusätzlich: Evaluationsfragebogen ab 1.2.2014 online
- ▶ **Vorgehen:**
 - ▶ Themenübersicht & Fragen der Studenten
 - ▶ bisherige Übungsaufgaben durchgehen
- ▶ **Am Ende: Themenvorschläge für letzte Vorlesung**

Behandelte Themen

Themen der Vorlesung & Übung

- ▶ **Einführung**
 - ▶ Programmiersprachen & C
- ▶ **Variablen und Datentypen**
 - ▶ Integer, Fließkomma
- ▶ **Arithmetik**
 - ▶ Operationen
 - ▶ Type-Casting
- ▶ **Ein und Ausgabe**
- ▶ **Programmfluss**
 - ▶ Bedingungen
 - ▶ Schleifen
- ▶ **Funktionen**
 - ▶ Grundlagen
 - ▶ Definition vs. Deklaration
 - ▶ Rekursion
 - ▶ Parameterübergabe
- ▶ **Arrays**
- ▶ **Zeiger**
 - ▶ & Variablen
 - ▶ & Arrays
 - ▶ & Funktionen
 - ▶ Arithmetik
- ▶ **Strings in C**
 - ▶ statisch vs. dynamisch
 - ▶ Arbeiten mit Strings
- ▶ **Zusammengesetzte Datentypen**
 - ▶ struct als Parameter
 - ▶ structs in structs
- ▶ **Konstanten**
- ▶ **dynamischer Speicher**
 - ▶ Speicherbereiche eines Programmes
 - ▶ malloc & sizeof & free
 - ▶ Probleme
 - ▶ dyn. Speicher für Strings, Strukturen und für Rückgabewerte
- ▶ **dynamische Datenstrukturen**
 - ▶ Kundenverwaltung, Liste
 - ▶ Implementierungsbeispiel

bisherige Übungsaufgaben

Besprechung von einzelnen Übungsaufgaben

Abschlussthemen

Vorschläge für die letzte Vorlesung

Was gibt es noch in C

- ▶ **Lesen und Schreiben von Dateien**
 - ▶ Stichworte: `fprintf`, `fscanf`, `fopen`, `fclose`, ...
 - ▶ Gehört zum Umfang der Standardbibliothek
- ▶ **mehr C-Präprozessor**
 - ▶ Anweisungen mit `#` wie
 - ▶ dient dem „Verwalten“ von Quellcode für z.B. verschiedene Systeme und einbinden von Bibliotheken
- ▶ **unions**
 - ▶ zusammengesetzter Datentyp, ähnlich der `struct`
 - ▶ Unterschied: alle Elemente einer Union benutzen selben Speicherbereich



Was gibt es noch in C

- ▶ Lesen und Schreiben von Dateien
- ▶ mehr C-Präprozessor
- ▶ unions
- ▶ Bitfields und Bitshifts
 - ▶ extra Datentypen zum setzen und lesen einzelner Bits
- ▶ Kompilieren mit Makefile
 - ▶ make ist skriptsprache zum bedingten Kompilieren
 - ▶ definieren von Abhängigkeiten z.B. von anderen Bibliotheken
 - ▶ automatisch neu-kompilieren von Teilen des Programmes bei Änderungen von Quellcode



Was gibt es noch in C

- ▶ Lesen und Schreiben von Dateien
- ▶ mehr C-Präprozessor
- ▶ unions
- ▶ Bitfields und Bitshifts
- ▶ Kompilieren mit Makefile
- ▶ Vom Programmcode zum Maschinencode
 - ▶ Details zu den einzelnen Schritten der Übersetzung eines Programmes



Was gibt es noch in C

- ▶ Lesen und Schreiben von Dateien
- ▶ mehr C-Präprozessor
- ▶ unions
- ▶ Bitfields und Bitshifts
- ▶ Kompilieren mit Makefile
- ▶ Vom Programmcode zum Maschinencode
- ▶ ANSI C-Standard
 - ▶ verschiedene C-Versionen und Unterschiede
 - ▶ K&R C, C-89/90, C-99



Was gibt es noch in C

- ▶ Lesen und Schreiben von Dateien
- ▶ mehr C-Präprozessor
- ▶ unions
- ▶ Bitfields und Bitshifts
- ▶ Kompilieren mit Makefile
- ▶ Vom Programmcode zum Maschinencode
- ▶ ANSI C-Standard
- ▶ komplexe Zahlen
- ▶ Debugging mit gdb (& eclipse) 15
- ▶ Test-Driven Development 10
- ▶ verfügbare Programmbibliotheken 15



**Welches Thema möchten
Sie in der letzten VL
hören?**

Vielen Dank!

- ▶ Bei Fragen einfach eine Mail an:
 - ▶ arvid@terzibaschian.de