

Programmierpraktikum C und C++

Feedback und Warm Up



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Roland Kluge

roland.kluge@es.tu-darmstadt.de

ES Real-Time Systems Lab

Prof. Dr. rer. nat. Andy Schürr

Dept. of Electrical Engineering and Information Technology

Dept. of Computer Science (adjunct Professor)

www.es.tu-darmstadt.de



BLOCK 1



Tag 1-Nachmittag: Anmerkungen

- In **Version 3 der VM** gibt es Probleme mit dem Aktualisieren der Git-Repos
 - **Neues Skript ist im Moodle**
 - **git pull** ist auf jeden Fall **nötig**, da V3 noch Folien von 2015 enthält.
- **Klimaanlage: Zugangstür bitte anlehnen**
- **Wir sind sehr dankbar für Feedback unter**
 - <https://github.com/Echtzeitsysteme/tud-cpp-exercises/issues> bzw.
 - <https://github.com/Echtzeitsysteme/tud-cpp-lecture/issues>
 - (Auch generelles Feedback)
- **Am Ende des Tages bitte...**
 - ... **einfach nur abmelden**, nicht den PC ausschalten.
 - ... **Ergebnisse sichern** – wir können leider für nichts außerhalb von %HOME% garantieren.
- Beschreibung von **Aufgabe 1.3** war fehlerhaft → **Aktualisiert im Übungsblatt!**



Vorlesungsunterlagen auf GitHub



Übungsunterlagen auf GitHub



Neues Skript zum Aktualisieren der Git-Repos



Link zum VM-Image



Tag 3: Rückschau und Warm Up (I)

Korrektur zu C++-Referenzen

- Konzeptionell: Referenzen sind ein Alias und haben keine eigene Adresse
- Implementierungstechnisch:
 - Referenzen können (natürlich!) Speicher belegen.
 - Beispiele:
 - Übergabe an bereits gelinkte Bibliotheken
 - Als Attribute von Klassen

Tag 3: Rückschau und Warm Up (II)



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Was ist hier verkehrt? Welches Problem wird auftreten?

```
// main.cpp
#include "functions.cpp"

int main() {
    myNewFunction();
    return 0;
}
```

```
//functions.hpp
int myNewFunction();
```

```
//functions.cpp
#include "functions.hpp"

int myNewFunction() {
    return -12;
}
```

* und &

- Welche **Bedeutung** kann der Asterisk (*) im Code annehmen?
- Welche **Bedeutung** kann das Ampersand (&) im Code annehmen?



Tag 3: Rückschau und Warm Up (III)

- Fragen für die **Evaluation** am letzten Tag
 1. Stärkerer Fokus auf C++(11) legen (= weniger Fokus auf C)?
 2. Selbsteinschätzung deiner C++-Kenntnisse (1=Keine Vorkenntnisse, 3=Eigene kleinere Projekte, 5=Sehr erfahrener C++-Programmierer)
- **Gastvortrag am letzten Tag:** Holger Wech von Cypress/Spansion



Holger Wech

Supervisor Microcontroller Application /
Spansion Inc.

Frankfurt am Main und Umgebung,
Deutschland | Halbleiter

Aktuell Spansion

Früher

Ausbildung Technische Universität Darmstadt



Tag 3: * und &

Welche **Bedeutung** kann der Asterisk (*) im Code annehmen?

Welche **Bedeutung** kann das Ampersand (&) im Code annehmen?



Tag 4: Rückschau und Aufwärmübungen

Übrigens: Nicht vergessen, hin und wieder mal zu pullen

- Beide Repositories
- **Eher kleine Bugfixes.**
- Foliennummerierung sollte stabil bleiben

Tag 5: Rückschau und Aufwärmübungen

Aufgabe T4A4.5 ist NICHT optional

- Fehler unsererseits – neue Version is online.
- Noch ein Grund, regelmäßig zu pullen.

Der Foliensatz für die zweite Woche wird zusätzliche Folien enthalten

- Siehe "Folien-Brutkasten" im aktuellen Foliensatz, 2016-09-15 in den Kommentaren

In der VM fehlen evtl. Header für die μ C-Projekte.

- Bitte innerhalb des ...exercises-Repo folgende Befehle aufrufen:
 - `git submodule init`
 - `git submodule update --recursive`

μ C-Beispielprojekte können nicht kopiert werden

- Nutzen relative Pfade im exercises-Repo
- Alternative 1: Kopieren und manuell das Makefile anpassen
- Alternative 2: Über "Create project" ein neues "FX16 project" anlegen.



Tag 6: Evaluationsfragen



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fragen für die Evaluation

- Stärkerer Fokus auf C++(11) legen (= weniger Fokus auf C)?
 - "trifft zu" = mehr C++
 - "Mitte" = gleich lassen
 - "trifft nicht zu" = mehr C
- Ich hatte vorher keine C++-Kenntnisse
 - "trifft zu" = Keine Vorkenntnisse
 - "Mitte" = Eigene kleinere Projekte
 - "trifft nicht zu" = Sehr erfahrener C++-Programmierer



Tag 6: Evaluationsfragen

Hinweise zur Klausur

- Wir gehen davon aus, dass **alle nicht-optionalen Übungen** der **Tage 1-4** bearbeitet wurden **und** alle Folien soweit verstanden wurden.
- Es wird ein **Codehandout** geben, in dem für bestimmte Funktionen/Datentypen/Methoden eine Beschreibung vorliegt.
 - NICHT zu erwarten ist bspw. `std::vector<T>::push_back`
 - Zu erwarten ist bspw. `std::string::find`





BLOCK 2

