

# Übungsblatt 1

## mit Lösung

Bearbeitung ab Montag, 27. April 2020

### 1.1 Grundlagen der praktischen Übungen

- Melden Sie sich mit Ihren Benutzerdaten an dem Testsystem an. Sie benötigen dazu einen SSH client (openssh in linux oder PuTTY in Windows). Melden Sie sich zuerst auf `ls12sq.cs.tu-dortmund.de` an (als gateway) um danach per `ssh ls12-epyc0.cs.tu-dortmund.de` das Testsystem zu erreichen. Machen Sie sich mit der Konsolenumgebung vertraut, sie sollten Verzeichnisse anzeigen, wechseln und erstellen können. Sie sollten ebenfalls Dateien anlegen, bearbeiten und löschen können.
- Untersuchen Sie das Testsystem auf seine Hardwareeigenschaften (Anzahl CPU Threads, Größe des Arbeitsspeichers). Hinweis: Das Linux Tool `lscpu` verrät ausführliche CPU Informationen.
- In den Übungen zu Rechnerarchitektur verwenden wir das Versionskontrollsystem **SVN**. Machen Sie sich mit SVN vertraut.  
**SVN-Repository:** <https://ls12-svn.cs.tu-dortmund.de/svn/courses-ss2020/ra2020/<username>>  
Die zur Lösung der Aufgaben benötigten Dateien finden Sie jeweils in den entsprechend benannten Ordnern. Legen Sie bitte eine "info.txt" Datei in dem Repository mit ihren Namen und Matrikelnummern an.
- Als C-Compiler verwenden wir **gcc**. Schreiben Sie ein einfaches 'Hallo Welt'-C-Programm. Übersetzen Sie es und führen Sie es aus.
- Suchen Sie nach Möglichkeiten, unter Linux die Laufzeit eines Programms zu messen, und probieren Sie diese mit Ihrer 'Hallo Welt'-Anwendung aus. Was fällt auf?

**Lösung:** Das programm `time` erlaubt externe Zeitmessungen: `time ./helloworld`  
`time` ist sehr 'ungenau' `perf` ist hier die beste Wahl:  
`perf stat ./helloworld`

### 1.2 Pthread-Programmierung

In der Betriebssysteme-Vorlesung haben Sie bereits **pthread**s kennen gelernt. In dieser Aufgabe soll nun eine pthread-Anwendung entwickelt werden. Dazu finden Sie im SVN-Repository eine sequentielle Programm (seq.c) in dem vier Funktionen aufgerufen werden. Ihre Aufgabe ist es, das Programm so umzuschreiben, dass diese vier Funktionen parallel ablaufen. Überprüfen Sie die korrekte Funktionsweise des parallelen Programms und messen Sie die Laufzeit.  
**Lösung:** Eine mögliche Lösung findet man unter `pthread.c`. Ein häufiger Fehler ist die `pthread_join` Funktion zu vergessen.

**Allgemeine Hinweise:** Informationen zur Veranstaltung finden Sie unter <https://ls12-www.cs.tu-dortmund.de/daes/de/lehre/lehrveranstaltungen/summersemester-2020/rechnerarchitektur-deutsch.html>. Eine Abgabe der Übungsblätter ist nicht erforderlich.