

Übungsblatt 1

Abgabe bis spätestens 2.11.2023 in Stud.IP

Aufgabe 0

Gebt auf Eurer Abgabe Euren Studiengang und das Fachsemester an (diese werden zu statistischen Zwecken erhoben und daher anonymisiert und über die Veranstaltung kumuliert an die Veranstalterin weitergegeben).

Gebt auf jeder Abgabe für jedes Gruppenmitglied die prozentuale Beteiligung am betreffenden Übungsblatt an. **Abgaben, bei denen diese Angabe fehlt, werden nicht korrigiert.**

Aufgabe 1 (4 Punkte)

In dieser Aufgabe geht es um das korrekte Zitieren von Literatur. Bitte beantwortet die folgenden Fragen und belegt Eure Antworten mit Textstellen aus einschlägigen Quellen in diesem Bereich. Verwendet mindestens ein wörtliches Zitat. Begründet, wieso Ihr diese Quellen für geeignet haltet.

- a) Was ist Moore's Law?
- b) Hat es heute noch Gültigkeit? Erörtert verschiedene in der Fachwelt diskutierte Standpunkte und gebt die Quellen dafür an (mindestens zwei).
- c) Welches Wachstum wird im Internet der Dinge (IoT) in den nächsten Jahren erwartet?

Bonus (1 Punkt): arbeitet für Frage b) mit einer anderen Gruppe zusammen. Diskutiert Eure Standpunkte. Achtet darauf, dass jede Gruppe ihre eigene Antwort erstellt, so dass Eure Eigenleistung erkennbar ist. Gebt an, mit welcher Gruppe Ihr zusammengearbeitet habt.

Aufgabe 2 (4 Punkte)

Gegeben sei das in Abbildung 1 skizzierte Dateisystem.

- a) Die Nutzerin `irma` führt nun nacheinander die folgenden Operationen aus. Geht davon aus, dass alle Operationen erfolgreich sind.

```
cd /home/ute/lehre/ti2-23
cp -r ueb2 ~/ti2/u2
ln -s `pwd`/ueb1 ~/ti2/u1
cd ~/ti2/u2
ls -la
cd ..
mv u2/Makefile .
cd -
ln ../Makefile
```

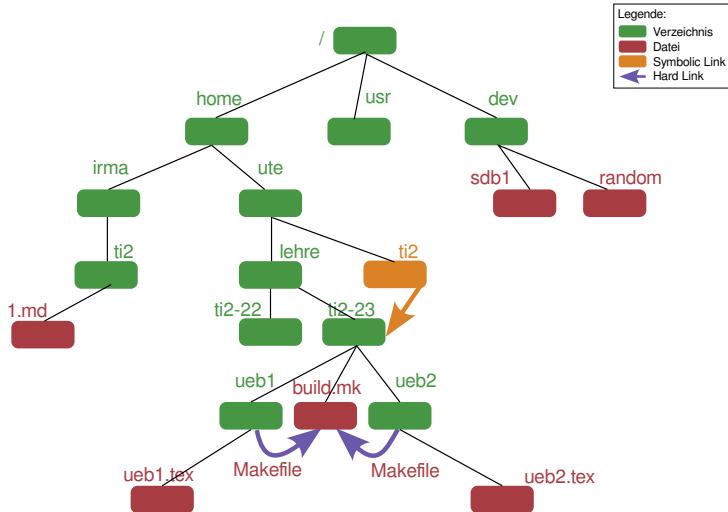


Abbildung 1: Dateisystem

Zeichnet am Computer den resultierenden Verzeichnisbaum, **nachdem** diese Änderungen ausgeführt sind und bindet das Ergebnis in Eure Abgabedatei ein. Ihr könnt als Ausgangsbasis die Vorgabedatei `dateibaum.svg` aus dem in StudIP zur Verfügung gestellten Archiv `ueb1-vorgabe.zip` verwenden. Falls Ihr kein Vektorgrafikprogramm (z. B. [inkscape](#)) zur Hand habt, könnt Ihr Euch notfalls mit Online-Diensten wie [draw.io](#) behelfen.

Wichtig: Gebt keine eingescannten, abfotografierten oder freihändig gemalten Zeichnungen ab — diese werden, wie in den Scheinbedingungen festgelegt, nicht gewertet.

- b) Anschließend führt die Nutzerin `irma` folgende Kommandos aus:

```
cd /home/irma/ti2
rm Makefile
```

Beschreibt, welche Änderungen im Dateisystem sich dadurch ergeben.
(Zeichnen ist nicht notwendig und auch nicht ausreichend.)

- c) Schließlich wird auch die Nutzerin `ute` aktiv:

```
cd ~/lehre
mkdir shared
mv ../ti2/build.mk shared
cd ../ti2
mv ueb1 u1
mv ueb2 u2
```

Zeichnet am Computer den resultierenden Verzeichnisbaum, **nachdem** alle Änderungen aus den Aufgabenteilen a–c ausgeführt sind und bindet das Ergebnis in Eure Abgabedatei ein.

Aufgabe 3 (2 Punkte)

In Unix-artigen Betriebssystemen werden alle Geräte wie Dateien behandelt. Dies betrifft auch den Standardeingabekanal `STDIN` und die Ausgabekanäle `STDOUT` (Standardausgabe) und `STDERR` (Fehler) eines Programms. Eine Benutzerin kann beim Aufruf eines Programms zum Beispiel entscheiden, ob die Eingabe aus einer Datei gelesen oder die Ausgabe in eine Datei

geschrieben werden soll. Erläutert jeweils für die folgenden Kommandofolgen, was dort im Hinblick auf die **Umlenkung von Eingaben und Ausgaben genau passiert**.

```
ls -d /etc/* > 1
ls `cat <1` 1>/dev/null
ls /etc/* 2>&1 | grep 'Permission denied'
```

Es wird vorausgesetzt, dass die aufgerufenen Kommandos verfügbar sind und sich im aktuell gesetzten Ausführungspfad befinden. Die Nutzerin besitzt Schreibrechte im aktuellen Verzeichnis, die umask hat den Wert 022, d. h. neu erzeugte Dateien sind ebenfalls lesbar und schreibbar. Das aktuelle Verzeichnis ist zu Beginn leer. Woher werden Eingaben gelesen und wohin Ausgaben geschrieben? Gibt es eine Umlenkung (wenn ja woher/wohin)? Wo werden neue Dateien erzeugt? Wo werden Ausgaben an bestehende Dateien angehängt?

Eine bloße Beschreibung der sichtbaren Auswirkungen auf Shell-Ebene oder der grundsätzlichen Funktionsweise der verwendeten Programme sowie generische Erläuterungen der Ein-/Ausgabeumlenkung („falls Schreibrechte bestehen, passiert...“) sind ausdrücklich nicht gefordert. Für die erfolgreiche Bearbeitung dieser Aufgabe muss zwingend beschrieben werden, **was in diesem konkreten Szenario hinsichtlich Pipes und Eingabe-/Ausgabeumlenkungen passiert**. Falsche Beschreibungen und Ungenauigkeiten können zu Punktabzug führen.

Abgabe

Bis 23:59 Uhr am 2.11.2023 digital in StudIP. Es gelten die vereinbarten Scheinbedingungen (siehe StudIP). Bitte beachtet unsere ergänzenden Hinweise ebenda.

Ladet die abgaberelevanten Dateien in DoIT hoch.

Eure Ansätze und der gewählte Lösungsweg müssen nachvollziehbar sein. Achtet insofern auf eine saubere Dokumentation im Quelltext und im abgegebenen PDF-Dokument. Benennt alle von Euch verwendeten Quellen, auch Zusammenarbeit mit anderen Gruppen und verwendete Unterlagen aus früheren Jahrgängen.

Programmieraufgaben sind im Zweifel in C++20 zu entwickeln. Die Korrektheit der Lösung bzw. deren Grenzen sind grundsätzlich in der Abgabe nachzuweisen. Dies geschieht neben der Dokumentation des Programmcodes in den Quelldateien und zusätzlich in dem abgegebenen PDF-Dokument mit der Lösungsbeschreibung durch geeignete Tests, deren Auswahl und Eignung begründet werden müssen. Als **Referenzplattform** gelten die Linux-Rechner im Rechnerpool in MZH E0.