

---

Aufgaben zur Klausur **Unix** im WS 95/96 (IA52)

Zeit: 60 Minuten

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen ausschließlich an den gekennzeichneten Stellen in das Aufgabenblatt ein. Ist ihre Lösung wesentlich umfangreicher, so überprüfen Sie bitte nochmals Ihren Lösungsweg.

Viel Erfolg !

Diese Klausur besteht einschließlich dieses Deckblattes aus 3 Seiten

---

### Aufgabe 1:

Schreiben Sie ein Shell-Skript `xcp` für die Bourne-shell `sh` (oder `bash` oder `ksh`), das zwei Parameter besitzt. Dieses Skript soll eine Datei (file) oder einen ganzen Verzeichnisbaum (directory) kopieren. Der erste Parameter des Skript bezeichnet die Quelle, der zweite das Ziel. Gibt es das Ziel schon, so soll es überschrieben werden, anderenfalls soll es neu angelegt werden.

Vor Ausführung des Aktionen sollen die gegebenen Parametern auf Korrektheit geprüft werden. Im Fehlerfall soll das Skript mit einem Exitcode ungleich 0 beendet werden.

Schreiben Sie das Skript so, daß es unabhängig von seinem Namen funktioniert.

Verwenden Sie das `cp` Kommando bitte nur für einfache Dateien, (bitte keine `-r` Optionen)

Hinweis:

`$1` bis `$9` sind die Namen der ersten 9 Parameter eines Skripts.

`$0` ist der Name des Skripts

`$#` gibt die Anzahl der Parameter an.

Operationen zum Testen:

[ `-f <datei>` ] einfache Datei

[ `-d <datei>` ] *directory*

[ `<arg1> -eq <arg2>` ]

[ `<arg1> -ne <arg2>` ]

[ `<arg1> -ge <arg2>` ]

[ `<arg1> -gr <arg2>` ] arithmetische Vergleiche

**Aufgabe 2:**

Gegeben seien folgende sechs C-Quellcode-Dateien `haupt.c`, `f1.c`, `f2.c`, `f1.h`, `f2.h` und `typen.h`. `haupt.c` enthält das Hauptprogramm, dies benutzt Routinen, die in `f2.h` deklariert sind und in `f2.c` implementiert sind. `f2.c` benutzt Routinen, die in `f1.h` deklariert sind und in `f1.c` implementiert sind. Alle `.c`-Dateien verwenden globale Daten-Definitionen aus `typen.h`.

Schreiben Sie einen `Makefile` zum Erzeugen eines Programms `prog` aus den oben beschriebenen Dateien. Beachten Sie dabei alle Abhängigkeiten zwischen den Dateien, entwickeln Sie den `Makefile` aber so, daß keine überflüssigen Aktionen gemacht werden.

Nutzen Sie keine im `make`-System vordefinierten oder eingebauten Regeln.