

Aufgabe 6:

printdir (12 Punkte) Bearbeitung in Zweier-Gruppen

Entwickeln Sie ein Programm **printdir**, mit dem Sie — ähnlich wie mit dem UNIX-Kommando **ls(1)** — den Inhalt eines oder mehrerer Verzeichnisse anzeigen können. Die Namen der Verzeichnisse werden auf der Kommandozeile übergeben.

a) Einfache Ausgabe

Das Programm soll auf der Standardausgabe für jedes Verzeichnis jeweils zuerst den Verzeichnisnamen und in den darauffolgenden Zeilen - durch einen Tabulator eingerückt - die Verzeichniseinträge ausgeben. Verzeichniseinträge, die mit einem Punkt beginnen, werden wie bei Unix-**ls** nicht angezeigt. Essentielle Funktionen: **opendir(3)**, **readdir(3)**

b) Makefile

Erstellen Sie ein zur Aufgabe passendes Makefile mit dem gleichen Funktionsumfang wie in der Aufgabe 2.

c) Sortieren der Ausgabe

Sortieren Sie die Ausgabe mit Hilfe von **qsort(3)** nach der Modifikationszeit (jüngste Datei zuerst ausgeben).

d) Dateityp, Zugriffsrechte, Dateigröße und Anzahl der Links

Geben Sie zum Dateinamen den Typ der Datei (regulär, block-special, ...), die Zugriffsrechte (rwx ...), die Dateigröße in Bytes und die Anzahl der Links (hardlinks) auf dies Datei aus. Verwenden Sie dazu den Systemaufruf **lstat(2)**.

Beispiel:

```
lrwxr-x--x  1 2038      stat.c
-rwxr-x--x  1 2038      realstat.c
```

e) Dateityp und Zugriffsrechte

Ihr Programm soll nun noch symbolische Verweise als solche anzeigen. (**readlink(2)**).

Beispiel:

```
lrwxr-x--x  1 10      stat.c -> realstat.c
-rwxr-x--x  1 2038      realstat.c
```

Hinweis zur Lösung dieser Aufgabe:

- Reagieren Sie in Ihrem Programm angemessen auf Fehler, wie etwa nicht lesbare Verzeichnisse.
- Verwenden Sie Systemmakros zur Auswertung der stat-Struktur und vom System vorgegebene Funktionen. Erfinden Sie diese nicht neu!
- Symbolische Links entsprechen nicht dem POSIX.1-Standard. Um die entsprechenden Funktionen benutzen zu können, müssen Sie Ihr Programm mit `-D_XOPEN_SOURCE=500` übersetzen.

Abgabe: bis spätestens Donnerstag, 14.06.2007, 12:00 Uhr