
SPiCAufgabe #6: printdir

(12 Punkte, Montag, 11.07.2011, um 8:00, keine Gruppen)

Entwickeln Sie ein Programm `printdir`, mit dem Sie - ähnlich wie mit dem UNIX-Kommando `ls(1)` - den Inhalt verschiedener Verzeichnisse angeben können.

Es wird empfohlen beim Entwurf des Programms in folgenden Arbeitsschritten vorzugehen:

- (a) Schreiben sie zunächst ein Programm, welches alle Einträge des aktuellen Ordner ausgibt (Je Eintrag eine Zeile). Einträge, deren Name mit einem `'.'` beginnen, sollen nicht angezeigt werden (versteckte Dateien). (`opendir(3)`, `readdir(3)`, `closedir(3)`)
- (b) Erweitern Sie das Programm nun so, dass vor dem Dateinamen, auch die Dateigröße ausgegeben wird. Der Name soll durch einen Tabulator (`'\t'`) getrennt werden. Am Ende der Ausgabe soll die Gesamtzahl der ausgegebenen Einträge, sowie deren Gesamtgröße ausgegeben werden. Ignorieren Sie außerdem alle Einträge, bei denen es sich nicht um eine reguläre Datei handelt. (`lstat(2)`)
- (c) Werten Sie nun die Parameter `argv` und `argc` aus. Alle übergebenen Parameter sollen als Verzeichnispfade interpretiert werden. Die Ausgabe eines Verzeichnisses soll mit `'<Verzeichnisname>:\n'` beginnen. Wird kein Parameter übergeben, soll das aktuelle Verzeichnis ausgegeben werden.

Hinweise:

- Sie finden im Verzeichnis `/proj/i4spic/pub/aufgabe6` für jede Teilaufgabe eine Beispiellösung, deren Ausgabe Sie mit Ihrer eigenen Lösung vergleichen können.
- Für die Abgabe ist nur die endgültige Lösung notwendig, die Sie in einer Datei `printdir.c` in Ihrem Projektverzeichnis ablegen sollen.
- Achten Sie auf aussagekräftige Fehlermeldungen, die alle auf dem Standardfehlerkanal ausgegeben werden sollen.
- Sie können davon ausgehen, dass Pfad und Dateiname maximal 1024 Zeichen lang sind¹. Achten Sie auch hier auf eine entsprechende Fehlermeldung.
- Die Funktionen der `string.h` können beim Lösen der Aufgabe hilfreich sein.

¹Alternativ kann auch `PATH_MAX` aus der `limits.h` verwendet werden.