

1. **cpf — Copy a File** Schreiben Sie ein einfaches Linux Programm in C, welches eine Datei in eine neue Zielfeile kopiert. Verwenden Sie die POSIX System Calls `open`, `creat`, `read`, `write` etc. anstelle der ANSI C Funktionen wie `fopen`, `fread` etc.

Das Programm `cpf` soll von der Command Line aus aufgerufen werden können:

```
cpf abc xyz
```

kopiert das File `abc` nach `xyz`. Falls `xyz` bereits existiert, soll die Datei überschrieben werden. Anderenfalls soll die Datei erzeugt werden. Das Programm muss mit genau zwei Parametern (zwei gültige Dateinamen) aufgerufen werden. Die erste Datei ist die Quelle, die zweite das Ziel.

2. **lsd — List directories** Schreiben Sie ein Programm `lsd`, welches alle Directories im aktuellen Arbeitsverzeichnis auf die stdout listet. Die Ausgabe Ihres Programms soll wie folgt angegeben, aussehen:

```
Peters-MacBook-Pro:lsd peter$ ./lsd
No directories!
Peters-MacBook-Pro:lsd peter$ mkdir bla
Peters-MacBook-Pro:lsd peter$ mkdir blu
Peters-MacBook-Pro:lsd peter$ ./lsd
bla blu
Peters-MacBook-Pro:051_inode peter$
```

3. **isdir — Check Directory** Schreiben Sie ein Programm `isdir` welches überprüft, ob das als Command Line Argument angegebene File ein Directory ist. Die Ausgabe des Programms soll so wie unten dargestellt aussehen. Falls das angegebene Command Line Argument nicht existiert, soll die Fehlermeldung "`usage ./isdir <filename>`" ausgegeben werden.

```
Peters-MacBook-Pro: 055_isdir peter$ ./isdir /dev/disk0s2
/dev/disk0s2 is no directory
Peters-MacBook-Pro: 055_isdir peter$ ./isdir /bla
/bla doesn't exist
Peters-MacBook-Pro: 055_isdir peter$ ./isdir local_dir
local_dir is a directory and contains files
Peters-MacBook-Pro: 055_isdir peter$ ./isdir
empty_dir empty_dir is a directory and is empty
Peters-MacBook-Pro:055_isdir peter$
```