

2. Übung zur Vorlesung „Betriebssysteme und Netzwerke“ (IBN)

Abgabedatum: 07.05.2019, 11:00 Uhr

Aufgabe 1

(4 Punkte)

In this exercise, you will create a script that organizes a folder full of images based on each image's creation date or last modified date. In details, the script should work as follows:

1. The script should receive a directory path as input. Any non-directory path should be rejected by the script with a proper error message.
2. The script should then inspect every `.jpeg` and `.png` file within this directory and create a subfolder per each year-month found in the files creation date or last modified date. Each subfolder should be named using the following format `YYYY-MM` (e.g., `2019-01`, `2019-02`).
3. Then, the script should move the files to their respective folder, based on each file creation date/last modified date.

For instance, running the script with the input of a directory `dir` containing the following files (filename, creation date/ last modified date):

```
dir/  
  
  pic1.jpeg 2019-03-21  
  pic2.jpeg 2019-03-28  
  pic3.png  2019-04-02  
  file.txt  2019-04-04
```

Should yield the following file structure:

```
dir/  
  
  file.txt 2019-04-04  
  2019-03/  
    pic1.jpeg 2019-03-21  
    pic2.jpeg 2019-03-28  
  2019-04/  
    pic3.png  2019-04-02
```

Aufgabe 2

(2 Punkte)

Linux verwendet neben *fork()* den Systemaufruf *clone()*, um neue Prozesse oder Threads zu erzeugen.

- a. Beschreiben Sie kurz die Parameter dieses Systemaufrufs. Beschreiben Sie die Unterschiede zu *fork()*.
- b. Der Parameter *int flags* erlaubt eine feine Abstufung, welche Ressourcen sich Elternprozess und Kindprozess teilen. Listen Sie die einzelnen Optionen auf und beschreiben Sie kurz deren Bedeutung. Bei welchem Wert von *flags* werden maximal viele Ressourcen gemeinsam verwendet (eine symbolische Angabe reicht)?

Aufgabe 3

(4 Punkte)

Erinnern Sie sich an den Befehl **fork()** und wie er funktioniert. Was bedeutet der Rückgabewert von **fork()**? Beantworten Sie folgende Fragen. Wenn Sie sich nicht sicher sind, oder Ihre Antwort einfach praktisch überprüfen wollen, fügen Sie eventuell Ausgaben oder **sleep(...)**-Befehle ein, kompilieren und starten Sie das Programm. In einer Bash-Shell können Sie sich mit dem Befehl **ps** über laufende Prozesse informieren.

Wie viele Prozesse (inklusive des 1. Elternprozesses) werden durch die folgenden beiden Programme jeweils erzeugt? Zeichnen Sie jeweils den Graph der entstehenden Prozesshierarchie, wobei dem Elternprozess das Label „1“ zugeordnet wird. Das erste Kind von Elternprozess „1“ sei dann „1.1“, das zweite „1.2“, usw.

Programm 1:

```
int main(int argc, char* argv[]) {
    int i=0;
    if (fork()!=0) i++;
    if (i!=0) fork();
    if (fork()!=0||i!=0) i++;
    if (fork()==0&& i==0) fork();
    return 0;
}
```

Programm 2:

```
int main(int argc, char* argv[]) {
    int i=0;
    if (fork()!=0) i+=2;
    if (fork()==0) i-=1;
    if (fork()!=0) i-=1;
    if (fork()*i==0) fork();
    return 0;
}
```

Aufgabe 4

(2 Punkte)

Bearbeiten Sie folgende Aufgabe in einer bash-Shell. Dazu können Sie ein eigenes System benutzen, oder sich beim URZ im Linux-Terminalserver¹ einloggen.

Speichern Sie folgendes C-Programm unter dem Dateinamen *test.c* und kompilieren Sie es (`gcc test.c -o test`).

```
#include <stdlib.h>
int main () {
    if (fork() > 0) { sleep(1000); }
    else { exit(0); }
    return 0;
}
```

Führen Sie nacheinander Folgendes aus:

- `./test &`
- `./test`
- Drücken Sie Strg+Z

Lassen Sie sich jetzt die Prozesshierarchie der Prozesse Ihrer Terminalsitzung anzeigen: `ps T -H`. Beschreiben Sie die Hierarchie und wie sie zustande kommt. Lassen Sie sich die gesamte Prozesshierarchie anzeigen (`ps -eH`). Auf welcher Tiefe im Baum findet Ihre Sitzung statt?

Schauen Sie in den man-pages zu `ps` nach, welche Prozesszustände vorliegen (Abschnitt „Process state codes“).

Aufgabe 5

(4 Punkte)

Schauen Sie das Video *Learning How to Learn - Introduction to Memory - Procrastination, Memory, and Sleep*². Hier werden zwei Arten des Gedächtnisses eingeführt und erklärt.

- Benennen und vergleichen Sie die beiden Arten des Gedächtnisses.
- Vergleichen Sie die beiden Gedächtnissysteme mit Speichersystemen in der Speicherhierarchie (Vorlesung 2). Betrachten Sie dabei die Aspekte Größe, Geschwindigkeit und Flüchtigkeit.
- Wie groß sind die beiden Gedächtnissysteme, d.h. wie viele Einheiten können sie jeweils enthalten und wie werden diese Einheiten im Video genannt?
- Es kann Information von beiden Systemen ins jeweils andere bewegt werden. Worin unterscheiden sich diese Richtungen? Für eine der beiden Richtungen wird im Video eine Technik erklärt. Benennen und beschreiben Sie sie.

¹<https://public.urz.uni-heidelberg.de/service-katalog/index.html?p=serverdienste/linux-terminalserver.html>

²<https://www.youtube.com/watch?v=0egaPpfFLI>