

1.5.A Anwendung/Prozesse – Arbeitsblatt

- Bearbeitungsdauer: 60 min
- Ergebnissicherung: Auf diesem Blatt
- Hilfsmittel: Lerntext, Man-Pages

Aufgabe 1: Liste der Prozesse

Mit den Programmen `ps`, `top` und `ps tree` kann man sich eine Liste der aktuell im System vorhandenen Prozesse ansehen.

- Welches ist auf Ihrem System die niedrigste PID, die der Befehl `ps ax` anzeigt?
- In der Spalte `COMMAND` sieht man, welches Programm zu einem Prozess gehört. Welches Programm gehört nun zu dem Prozess, der auf Ihrem System die niedrigste PID hat?
- Woher holt sich der Befehl `ps` eigentlich seine Informationen (Hinweis: `man ps`)?

Aufgabe 2: Signale

In einem Linux-System gibt es für Prozesse viele durchdachte Möglichkeiten, miteinander zu kommunizieren. Eine sehr einfache Möglichkeit ist das Senden von Signalen an andere Prozesse.

- Wie lautet die Option des Programms `kill`, mit der man sich eine Liste der möglichen Signale anzeigen lassen kann?
- Wie viele Signale sind auf Ihrem System möglich?
- Anwendungsentwickler können prinzipiell jedes Signal für eigene Zwecke benutzen. Aber welche Signale sind speziell dafür freigehalten?

Aufgabe 3: Job-Control

Die Shell ermöglicht es, Prozesse im Hintergrund zu starten und einen Prozess zwischen Vordergrund und Hintergrund zu verschieben. Zum Erlernen dieser Möglichkeiten bietet es sich an, das Programm `sleep` zu verwenden, das einfach nur eine bestimmte Anzahl von Sekunden abwartet, bevor es sich beendet.

- Wenn man mehrere Prozesse im Hintergrund startet, bekommt man Ausgaben wie:

```
[1]+  Fertig                sleep 5
```

Was bedeutet das Pluszeichen? Und was bedeutet ein Minuszeichen an dieser Stelle?

- Welche Befehlszeile muss man eingeben, um folgende Ausgabe zu bekommen:

```
[1] Fertig          sleep 4
[2]- Fertig         sleep 3
[3]+ Fertig         sleep 2
```

Befehlszeile:

Aufgabe 4: Signale und Job-Control

Die Shell verwirklicht die Job-Control unter anderem mit Hilfe von Signalen. Aber auch andere Prozesse und Skripte können mit Signalen Ähnliches bewirken:

- Mit welchem Signal kann man einen laufenden Prozess in den Zustand *angehalten* bringen?
- Mit welchem Signal kann man einen angehaltenen Prozess wieder zurück in den Zustand *laufend* bringen?
- Wenn man einen angehaltenen Prozess durch das richtige Signal wieder zurück in den Zustand *laufend* bringt, läuft er dann im Vordergrund oder im Hintergrund?

Aufgabe 5: Prioritäten

Sie sollen den Befehl `sleep 30` auf verschiedene Arten starten. Geben Sie bitte die Werte für PR und NI bei den folgenden Fällen ein:

- Start des Befehls im Vordergrund ohne `nice`
- Start im Hintergrund ohne `nice`
- Start mit `nice -n 3`
- Start mit `nice -3`
- Start ohne `nice`, anschließend Änderung mit `renice -n 5`
- Start ohne `nice`, anschließend Änderung mit `renice +5`

Aufg.	Befehlszeile	PR	NI
a)			
b)			
c)			
d)			
e)			
f)			