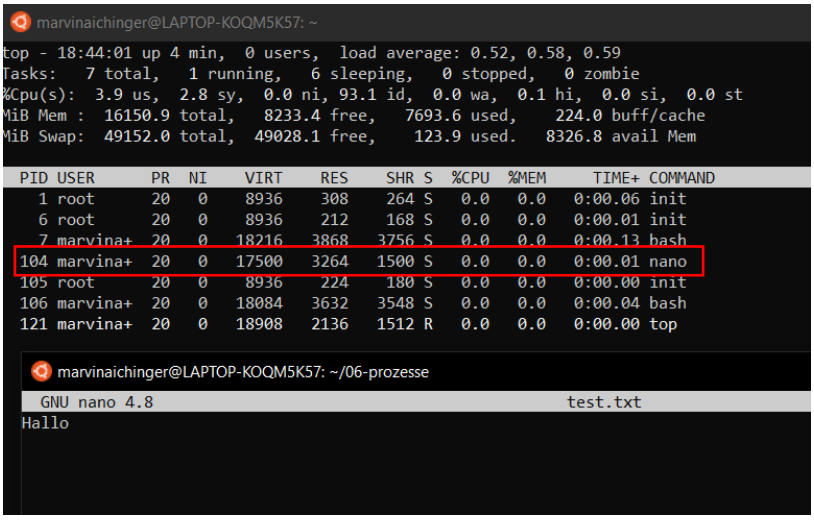
Prozesszustände

Prozesszustände in Linux:

1/2)

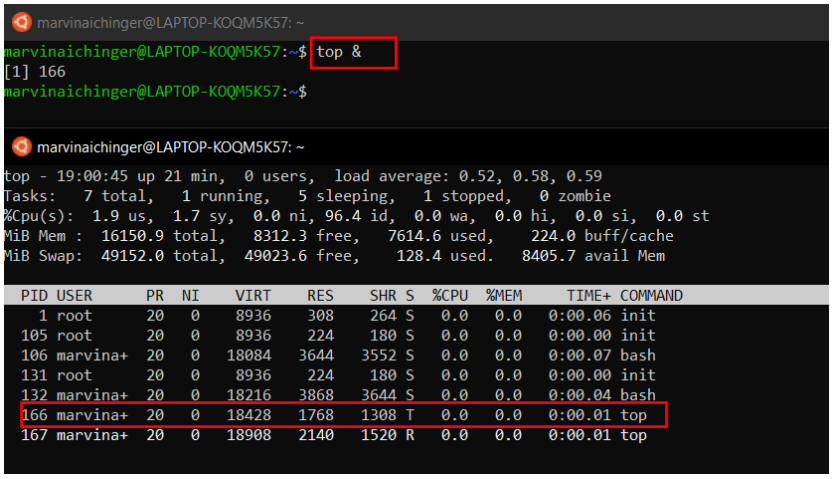



```
marvinaichinger@LAPTOP-KOQM5K57:~$ ps -aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.0   8936   308 ?        Ssl  18:39   0:00 /init
root           6  0.0  0.0   8936   212 tty1     Ss   18:39   0:00 /init
marvina+       7  0.0  0.0  18216  3868 tty1     S    18:39   0:00 -bash
marvina+     104  0.0  0.0  17500  3264 tty1     S    18:43   0:00 nano test.txt
root         105  0.0  0.0   8936   224 tty2     Ss   18:43   0:00 /init
marvina+     106  0.0  0.0  18084  3632 tty2     S    18:43   0:00 -bash
marvina+     126  0.0  0.0  18880  2032 tty2     R    18:45   0:00 ps -aux
marvinaichinger@LAPTOP-KOQM5K57:~$ ps -aux | grep nano
marvina+     104  0.0  0.0  17500  3264 tty1     S    18:43   0:00 nano test.txt
marvina+     128  0.0  0.0  16208  1288 tty2     S    18:46   0:00 grep --color=auto nano
marvinaichinger@LAPTOP-KOQM5K57:~$ ps
  PID TTY          TIME CMD
  106 tty2     00:00:00 bash
  129 tty2     00:00:00 ps
marvinaichinger@LAPTOP-KOQM5K57:~$ pstree
init─┬─init───bash───nano
     ├─init───bash───pstree
     └─{init}
marvinaichinger@LAPTOP-KOQM5K57:~$
```

3)

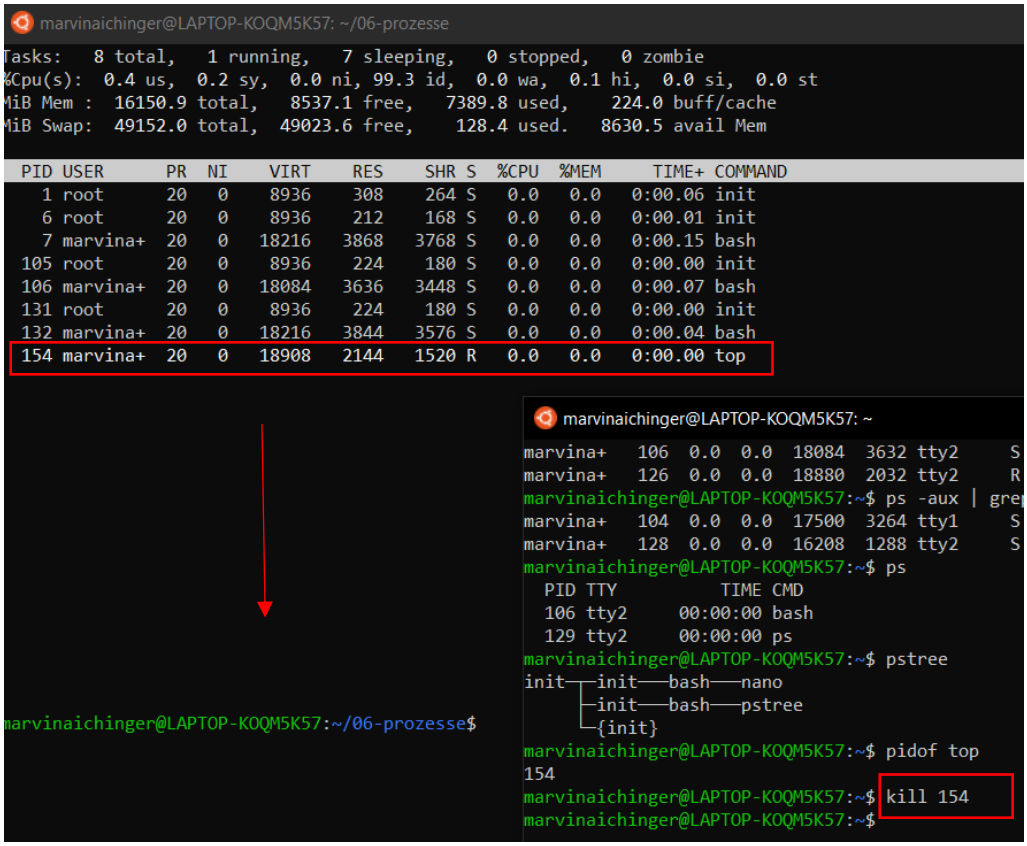
```
156 marvina+  20   0   18916   2148   1520 T   0.0   0.0   0:00.01 top
159 marvina+  20   0   18908   2144   1520 R   0.0   0.0   0:00.01 top
```

4)



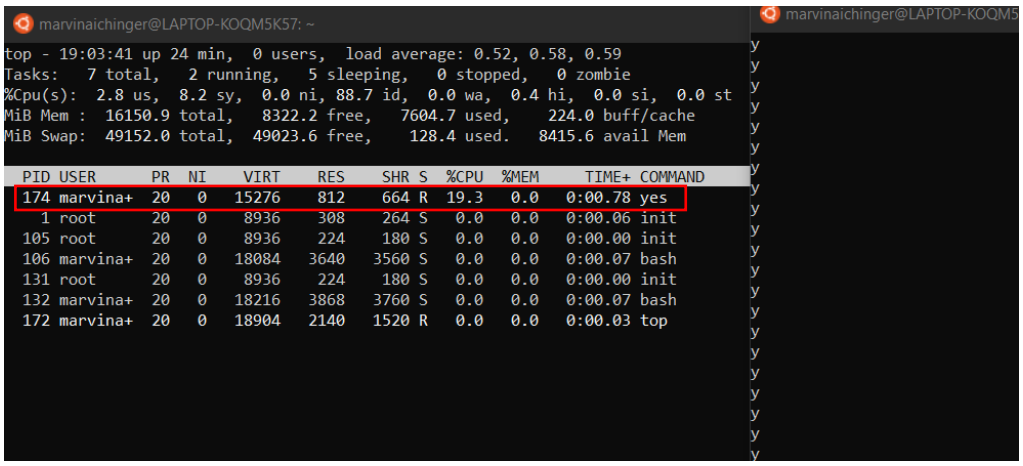
5)

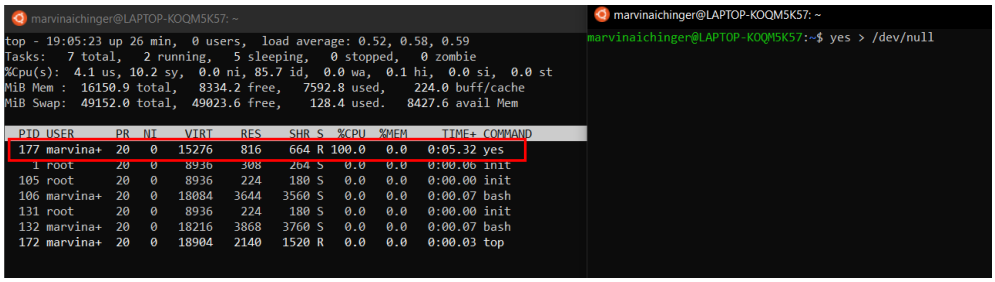
Mit dem Befehl „fg 1“ kann ein Prozess im Hintergrund wieder in den Vordergrund gebracht werden. Dabei ist immer die Nummer anzugeben, welche beim Starten in den Hintergrund in den [] - Klammern ausgegeben wird. Mit STRG+C kann der Prozess dann beendet werden. Auch möglich ist es den Prozess mit *kill* zu beenden:



6)
Ein Prozess muss nicht ständig Daten verarbeiten oder Berechnungen durchführen. Oft ist es für einen Prozess nötig seine Arbeit niederzulegen und darauf zu warten, bis ein bestimmtes Ereignis passiert. Zum Beispiel, dass eine bestimmte Datei oder irgendwelche Daten aus dem Internet angekommen sind oder, dass der User eine bestimmte Eingabe gemacht hat, usw.

7)





Der Prozess yes ist fast ständig im Running betrieb, weil er immer wieder eine while-Schleife durchgeht. Wenn die ganzen y's auf /dev/null umgeleitet werden, muss der Prozess nicht einmal darauf warten bis, dass die Ausgabe gemacht wurde, deshalb hat er dann eine CPU-Auslastung von 100%.

Linux Prozesspriorität:

- Eine Prozesspriorität wird verwendet, um jedem Prozess die passende CPU-Leistung zuzuordnen. Der nice-Level ist ein Wert, der vom User gesetzt werden kann, um die Priorität eines Prozesses zu verändern. Dabei sind werte von -20 bis 19 möglich mit -20 als am meisten wichtig. „Nicht-Root-User" haben nur die Möglichkeit Werte von 0 bis 19 zu setzten. (Priority = NiceLevel + 20)
- Beim Starten kann mit dem Code „nice -n <value> <command>" ein Befehl mit einem Nice-Level gestartet werden. (z.B.: nice -n 10 yes)
  Während der Befehl läuft kann mit dem Code „renice -n <value> -p <PID>" der nice-Level eines Prozesses (PID) verändert werden. (z.B.: renice -n 20 -p 156)
- Wenn nur eine 1-Kern CPU zur Verfügung steht und zwei Prozesse laufen, welche viel CPU-Leistung benötigen (z.B.: yes > /dev/null) dann wird die Leistung gerecht aufgeteilt. Das bedeutet, dass in diesem Fall jeder yes-Befehl 50% der CPU-Leistung bekommt. Falls aber es aus irgendeinem Grund nötig ist, dass ein yes-Befehl mehr Leistung bekommen soll, kann mit einem geringerem Nice-Level dies erreicht werden.