

Aufgabe 1)

1. Ein Prozess ist ein Programm das aktiv ist
2. Der PCB (process control block) auch Prozessor-kontrollblock genannt, beinhaltet viele Informationen um ein Prozess zu kontrollieren.

Ein PCB pro Prozess.

Zu den Einträgen gehören:

Identifier: Dies ist ein key, der zur Erkennung/Identifikation eines Prozesses dient. Dieser key ändert sich während der Laufzeit nicht

Rechenstatus (state): kann rechenbereit, rechnend, blockiert oder beendet annehmen

Befehlszähler (program counter): enthält die Adresse vom Befehl, das als nächstes ausgeführt werden soll.

Scheduling-Information: Hier gehört die Priorität hin, sowie der Zeiger

Priorität (priority): Entscheidet welcher Prozess Vorrang hat.

Memory management: Enthält Werte der Register sowie Pointer auf den Code und die Daten sowie auf den Stack

Accounting Information: Das Betriebssystem merkt sich wie lange ein Prozess aktiv ist und wie viel Speicher er belegt.

I/O - Status: Beinhaltet alle I/O<sup>geräte</sup>, die mit dem Prozess verbunden sind.

- Enthält auch Elternprozess, Prozessgruppe, CPU-Zeit der Kinderprozesse, sowie Zugriff und Benutzerrechte.

3) Ein Process hat einen eigenen Speicherbereich, während sich alle in selben Process gehörenden Threads sich den Speicherbereich teilen.

Anmerkung: Wir haben angenommen, dass nachdem ein Prozess gelinkt wurde, ready kommt.

Prozess zeit	P <sub>1</sub>	P <sub>2</sub>	P <sub>3</sub>	P <sub>4</sub>
0	new			
1	running	new		
2	running	ready	new	
3	running	ready	ready	new
4	running	ready	ready	ready
5	waiting	running	ready	ready
6	wait	running	ready	ready
7	wait	killed	running	ready
8	wait	ready	blocked	blocked
9	wait	running	blocked	blocked
10	wait	running	blocked	blocked
11	ready	running	wait	blocked
12	ready	running	wait	blocked
13	running	blocked	wait	blocked
14	running	blocked	wait	blocked
15	running	blocked	wait	blocked

16	running	blocked	ready	wait
17	terminated	blocked	running	wait
18		blocked	running	wait
19		blocked	running	wait
20		blocked	terminated	wait
21		wait		ready
22		wait		running
23		wait		running
24		wait		running
25		wait		running
26		ready		running
27		ready		terminated
28		running		
29		running		
30		running		
31		running		
32		terminated		

b) bcsg waiting passiert während des running, da man darauf kein anderen Prozess ausführen kann und der Prozess wartet während man eine while Schleife ausführt.

c) Es ist sinnvoll Prozesse abbrechen, wenn sie in eine Dauerschleife geraten. Unter UNIX kann man die Prozesse mit `Strg + C` abbrechen.