

Hausaufgabe 2

Gruppenmitglieder:

Vincent Wehr

Yannek Wehr

Andreas Dyballa

Junyi Chen

2.3

a)

Ein Prozess ist eine Instanz eines Programmes. Er ist spezialisiert auf deren Ausführung.

- besitzt Ressourcen
- ist eine Verarbeitungsvorschrift
- Aktivitätsträger

Durch Prozesse können die Ausführungen verschiedener Programme separiert werden. Des weiteren können durch diese Trennung logische Abhängigkeiten dargestellt und aufgelöst werden.

Quelle: Folie 2,3

b)

Nebenläufigkeit: Logisch simultane Verarbeitung von Operationsströmen
(wird durch Sequentialisierung gelöst)

Parallelität: Tatsächlich simultane Verarbeitung von Operationsströmen z.B. auf mehreren Prozessoren

Quelle: Folie 12, 13

c)

Explizites Umschalten: Prozess gibt die CPU ab

Implizites Umschalten: Prozess löst einen blockierenden Systemaufruf aus

Quelle: Folie 17

Bei einer Prozessumschaltung werden die Registerinhalte in den PCB gespeichert und der Prozesszustand wird aktualisiert. Danach wird ein anderer Prozess geladen. Dieser erhält sein passenden virtuellen Adressraum aus dem PCB, somit werden seine Registerinhalte geladen und der Prozess wird, durch den gespeicherten Befehlszähler, an der Stelle fortgesetzt wo dieser unterbrochen wurde.

Quelle: Folie 18

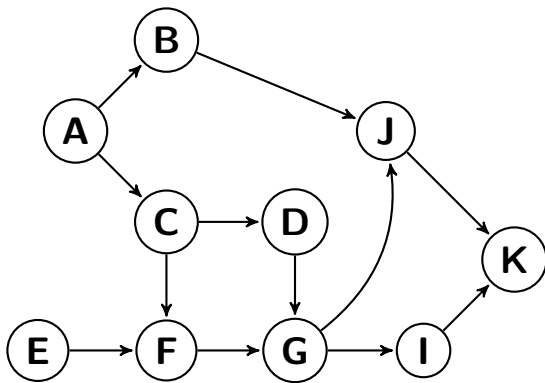
d)

- Threads sind der ausführende Teil eines Prozesses, in Form eines Kontrollflusses des Prozesses.
- Prozesse benutzen separate virtuelle Adressräume und können sich somit nicht beeinflussen.
- Threads benutzen den selben virtuellen Adressraum.
- Beim Umschalten wird der Ausführungszustand eines Prozesses im PCB gespeichert, bei Prozessen hingegen in einer Threadtabelle.
- Threads werden von dem startenden Prozess verwaltet, Prozesse vom Kernel.

Quelle: Folie 25, 26, 33, 34

2.4

a)



H

b)

```
fork k:=jobH()
fork e:=jobE()
  a:=jobA()
fork b:=jobB(a)
  c:=jobC(a)
join e:=jobE()
fork f:=jobF(c,e)
  d:=jobD(c)
join f:=jobE(c,e)
  g:=jobG(d,f)
join b:=jobB(a)
fork j:=jobJ(b,g)
  i:=jobI(g)
join j:=jobJ(b,g)
  k:=jobK(i,j)
join k:=jobH()
```