

Betriebssysteme und Sicherheit, WS 2022/23

2. Aufgabenblatt – Prozesse und Parallelität

Bearbeitungszeitraum: 24.10.2022 – 28.10.2022

Prozess (nach Nehmer): dynamisches Objekt, das selbständige, voneinander isolierte sequentielle Aktivitäten bezüglich Anfordern und Besitz von Betriebsmitteln in einem Rechensystem repräsentiert.

Ein Prozess ist gegeben durch:

Adressraum: Behälter für die Aufnahme abgegrenzter Programme mit zugeordneten Daten, Abstraktion des physischen Speichers

Handlungsvorschrift oder Code: im Adressraum in Form eines sequentiellen Programms gespeichert

Aktivitätsträger oder Thread: führt Handlungsvorschrift aus, Abstraktion des physischen Prozessors

Aufgabe 2.1 Prozesse können sich während ihrer Ausführung in einem von mehreren Zuständen befinden. Vollziehen Sie die Zustände eines Prozesses während seiner Ausführung anhand der gegebenen Beispielschritte nach.

1. Erzeugen eines neuen Prozesses
2. Starten des Programms in dem Prozess
3. Warten auf Eingabe des Nutzers
4. Eingabe des Nutzers
5. Berechnung
6. Ausgabe des Ergebnisses
7. Beendigung des Programms und Prozess

Welche weiteren Zustände sind noch möglich während der Ausführung eines Prozesses?

Aufgabe 2.2 Betrachtet werde ein Mail-Server, der in einer Liste vorliegende Nachrichten an den jeweiligen Empfänger versenden soll; jede Nachricht habe nur einen einzigen Empfänger. Der Server arbeite gemäß dem nebenstehenden Pseudocode. Dabei werde angenommen, dass stets zu versendende Nachrichten vorliegen.

```
while(true) {
    Nachricht <- Entnehmen(Liste)
    Empfänger <- Extrahieren(Nachricht)
    IP <- IP_Ermitteln(Empfänger)
    Verbindung <- Verbindung_aufbauen(IP)
    Senden(Nachricht, Verbindung)
    Verbindung_schließen(Verbindung)
}
```

- (a) Warum ist diese Lösung ineffizient? Welche Möglichkeiten gibt es, sie zu verbessern? Bietet sie dennoch Vorteile?
- (b) Beschreiben Sie eine Implementation, bei der mehrere Threads innerhalb eines Prozesses genutzt werden können. Welche Vor- und Nachteile hat dieses Vorgehen?
- (c) Welche Konsequenzen hat es, wenn anstelle der Threads selbst wieder sequentielle Prozesse verwendet werden?

Aufgabe 2.3 Implementieren Sie ein C-Programm, das die Summe einer Zahlenfolge parallelisiert mittels mehrerer Threads berechnet. Informieren Sie sich dazu über die Funktionsweise von `pthread_create` und `pthread_join`.