

# Betriebssysteme und Rechnernetze

Klausur WiSe 18/19

13. März 2019

*Gesamtpunktzahl: 30 Punkte*

## Aufgabe 1: Scheduling

- a) Entscheiden Sie für jedes der folgenden Schedulingverfahren, ob ein Verhungern einzelner Prozesse möglich ist? Begründen Sie Ihre Antwort!
- First Come First Served (FCFS - non-preemptive)
  - Shortest Job First (SJF - non-preemptive)
  - Round Robin (RR - preemptive)
  - Round Robin mit Prioritäten (preemptive)
- b) Gegeben sind 5 Prozesse  $P_1$  bis  $P_5$ . Die Prozesse sind zum jeweils angegebenen Zeitpunkt in der Reihenfolge ihrer Prozessnummern eingetroffen. Sie werden nach Prozessen, die zum angegebenen Zeitpunkt verdrängt worden sind, in die Warteschlange eingeführt. Ankunfts- und Rechenzeit sind in der folgenden Tabelle aufgeführt:

Prozesse	Ankunftszeit	Rechenzeit
$P_1$	0	8
$P_2$	1	4
$P_3$	5	3
$P_4$	9	6
$P_5$	13	5

Bestimmen Sie für Round Robin (RR) mit einer Zeitscheibe von 2 Zeiteinheiten die folgenden Kenngrößen:

- i) mittlere Verweilzeit
- ii) mittlere Wartezeit

Geben Sie das zugehörige Gantt-Diagramm an!

**Hinweis:** Die Zeit für den Prozesswechsel soll vernachlässigt werden. Alle Prozesse verbrauchen ausschließlich CPU-Zeit und werden nicht durch Ein- bzw. Ausgabeoperationen blockiert!

(2 + 2 Punkte)

## Aufgabe 2: Prozesse

- a) Ordnen Sie die Interprozesskommunikationsarten Shared Memory, Pipes und Sockets den Eigenschaften zu:
- **arbeiten nachrichtenbasiert:**
  - **arbeiten speicherbasiert:**
  - **arbeiten über Rechengrenzen:**
  - **funktionieren bidirektional:**
- b) Welches Verfahren der IPC würden Sie für die folgenden Problemstellungen vorschlagen? Begründen Sie ihre Entscheidung!
- i) Datenaustausch auf einer Mehrprozess-Maschine zwischen Prozessen, die nicht miteinander verwandt sind.
  - ii) Aufwecken eines unterbrochenen Prozesses
- c) Ein Elternprozess (PID = 42) mit den in der Tabelle gegebenen Eigenschaften erzeugt mit Hilfe des Systemaufrufes `fork()` einen Kindprozess (PID = 127). Tragen Sie die fehlenden Werte in die Tabelle ein!

	Elternprozess	Kindprozess
UID	11	
PPID	36	
PID	42	127
<code>fork()</code> Rückgabewert		

(2 + 2 + 2 Punkte)

## Aufgabe 3: Speicherverwaltung

- a) Ermitteln Sie die Anzahl der Seitenfehler für die Ersetzungsstrategien LRU und Working-Set mit dem Working-Set-Parameter  $T = 2$  für Seiten des virtuellen Speichers mit folgenden Seitenzugriffen:

5 - 3 - 2 - 4 - 1 - 1 - 2 - 3 - 4 - 3 - 5

Es wird von einem Speicher mit 4 Kacheln (Rahmen) ausgegangen.

**Hinweis:** In denjenigen Fällen in denen Nicht-Determiniertheit vorliegt ist stets die Seite mit der kleinsten Kachelnummer zu wählen, die das jeweilige Verfahren erfüllt.

- b) Wie wird die Working-Set-Strategie in realen Betriebssystemen approximiert?
- c) Benutzen globale Seitenersetzungsstrategien globale Seitentabellen? Begründen Sie ihre Antwort!
- d) Kann es bei LRU zu interner Fragmentierung kommen? Begründen sie!

(2 + 1 + 1 + 1 Punkte )

## Aufgabe 4: Synchronisation

- a) An der Mündung eines Kanals in einen Fluss befindet sich eine Schleuse. Aus Sicherheitsgründen darf höchstens ein Schiff die Schleuse befahren.
- i) Entwerfen Sie eine Semaphorlösung, die die Sicherheitsrichtlinie erfüllt!  
Geben Sie die Lösung in Pseudocode an!
  - ii) Wegen zunehmenden Schiffsverkehr wird die Schleuse ausgebaut und es existieren nun zwei Schleusenkammern. Die Sicherheitsrichtlinie gilt weiterhin für jede einzelne Schleusenkammer. Entwerfen Sie eine Semaphorlösung für diese Situation oder modifizieren Sie Ihre Lösung von Teil i)! Geben Sie die Funktionen `schleuse_einfahren` und `schleuse_ausfahren` in Pseudocode an!
- b) Zwei Programme  $P_1$  und  $P_2$  nutzen gemeinsam zwei Semaphore. Die Initialisierung und Teile der Programme sind in Pseudocode gegeben:

```
int count_1, count_2;  
Semaphore mutex_1 = 1, mutex_2 = 1;
```

P1:	P2:
DOWN(mutex_1);	DOWN(mutex_2);
count_1++;	count_2++;
DOWN(mutex_2);	DOWN(mutex_1);
count_2--;	count_1--;
UP(mutex_2);	UP(mutex_1);
UP(mutex_1);	UP(mutex_2);

Welche Anomalie kann hier auftreten und wie muss der Pseudocode korrigiert werden, um diese Anomalie zu beheben?

((2 + 1) + 2 Punkte)

## Aufgabe 5: Sicherungsschicht

- a) Um Übertragungsfehler zu korrigieren werden Datenworte mit Hilfe des Hamming-Codes in ein Codewort überführt.
- Gegeben sei folgendes Datenwort: 01001101. Berechnen Sie den zu diesem Wort gehörenden Hamming-Code, welcher dann übertragen wird! Geben Sie den Rechenweg an!
  - Ein Empfänger erhält das Codewort 0101 0110 0011. Wie heisst das empfangene Datenwort? Hat der Empfänger das Datenwort richtig erhalten? Begründen sie!
  - Wie viele Prüfbits werden im Hamming-Code benötigt, um jeden 1-Bit Fehler eines  $m$  Bit langen Datenworts beheben zu können?
- b) Geben Sie die Headerfelder vom HDLC-Protokoll an! Wie groß ist der Header?

((1 + 1 + 1) + 1 Punkte)

## Aufgabe 6: Rechnernetze

- a) Beschreiben Sie den prinzipiellen Übertragungsablauf beim Stop-and-Wait-Protokoll! Begründen sie, warum hier eine 1-Bit-Sequenznummer gebraucht wird!
- b) Welche Form der Flusskontrolle gibt es im Ethernet-Standard? Warum können trotzdem Pakete verloren gehen?
- c) Gegeben sei die folgende Wegetabelle eines Linux-Routers

Zielnetz-IP	Gateway	Netzmaske	Interface
192.168.1.0	0.0.0.0	255.255.255.0	wlan0
192.168.2.0	0.0.0.0	255.255.255.0	eth1
141.89.59.0	0.0.0.0	255.255.255.0	eth0
default	141.89.59.254	0.0.0.0	eth0

Der Linux Router besitzt 3 Schnittstellen wlan0, eth0 und eth1.

- Über welches Interface werden Pakete mit der Ziel-IP 141.89.59.42 weitergeschickt?
- Über welches Interface werden Pakete mit der Ziel-IP 141.89.64.1 weitergeschickt?

Wessen MAC-Adresse wird in dem jeweils zugehörigen Ethernet-Rahmen eingetragen? Begründen Sie ihre Antwort!

(2 + 2 + 2 Punkte)