

Betriebssysteme im Wintersemester 2017/2018

Übungsblatt 4

Abgabetermin: 20.11.2017, 18:00 Uhr

Besprechung: Besprechung der T-Aufgaben in den Tutorien vom 13. – 17. November 2017
Besprechung der H-Aufgaben in den Tutorien vom 20. – 24. November 2017

Aufgabe 16: (T) Betriebsmittel

(– Pkt.)

Eine der wesentlichen Aufgaben eines Betriebssystems ist die Verwaltung von Betriebsmitteln.

- a. Finden Sie für die folgenden Klassen von Betriebsmitteln jeweils Beispiele aus dem Hardware- oder aus dem Software-Bereich:
 - (i) einmalig benutzbar
 - (ii) wiederholt benutzbar
 - (iii) parallel benutzbar
 - (iv) unterbrechbar
 - (v) nicht unterbrechbar
- b. Bei der Betriebsmittelverwaltung sind eine Reihe von strategischen Entscheidungen zu treffen. Geben Sie einige Problemkreise an, in denen solche Entscheidungen anfallen, und geben Sie Strategien bzw. Mechanismen zu deren Lösung an.

Aufgabe 17: (T) Unix Prozesse

(– Pkt.)

In dieser Aufgabe sollen Sie sich mit der Verwendung der Systemfunktion `fork()`¹ vertraut machen. Betrachten Sie dazu das folgende C-Programm

```
1 #include <unistd.h>
2 #include <stdio.h>
3 #include <sys/types.h>
4
5 int main(void) {
6     pid_t kind = fork();
7
8     execl("./hallowelt", " ", NULL);
9     return 0;
10 }
```

¹Informationen zur Systemfunktion `fork()` erhalten Sie in der entsprechenden Manualpage (`man fork`).

Das Bash-Skript `hallowelt` beinhalte folgenden Code:

```
1 #!/bin/bash
2 echo Hallo Welt
```

Beantworten Sie nun die folgenden Fragen.

- Welche Ausgabe liefert obiges C-Programm auf einem Unix-System? Erklären Sie die Ausgabe.
- Sie werden beauftragt ein Terminal für Unix zu programmieren, in welches der Benutzer den Namen eines Programmes eingeben kann, welches dann ausgeführt wird. Wird das Programm gestartet, soll der Benutzer weiter Eingaben machen können. Erklären Sie, welche wesentlichen Systemaufrufe Sie wie verwenden müssen.
- Wie hängt Ihr Programm mit der Baumstruktur von Prozessen in Unix zusammen?
- Was passiert, wenn der Terminalprozess stirbt?

Aufgabe 18: (T) Dispatching von Prozessen

(– Pkt.)

Angenommen es existieren drei Benutzer-Prozesse, deren Spuren (Traces) wie folgt aussehen:

Trace des Prozesses A	Trace des Prozesses B	Trace des Prozesses C
$\alpha + 0$	$\beta + 0$	$\gamma + 0$
$\alpha + 1$	$\beta + 1$	$\gamma + 1$
$\alpha + 2$	$\beta + 2$	$\gamma + 2$
(E/A-Op.)	$\beta + 3$	$\gamma + 3$
	$\beta + 4$	(E/A-Op.)
	$\beta + 5$	
	$\beta + 6$	
	$\beta + 7$	
	$\beta + 8$	
	$\beta + 9$	
	$\beta + 10$	

Dabei sind α , β und γ die Anfangsadressen der Prozesse A bzw. B und C. Bei Prozess A soll angenommen werden, dass der dritte Befehl (also $\alpha + 2$) eine E/A-Operation ist, die ein Warten des Prozesses bedingt. Ähnliches gilt für Prozess C und dessen vierten Befehl $\gamma + 3$.

Welche Sequenz abzuarbeitender Befehle ergibt sich aus Sicht des Prozessors, wenn die Ausführung eines Prozesses nach maximal sechs Befehlszyklen durch ein Timeout unterbrochen wird und darauffolgend ein Dispatcher-Prozess für weitere sechs Befehlszyklen aktiv wird, der die Kontrolle an den nächsten Benutzer-Prozess übergibt? Die Befehle des Dispatcher-Prozesses befinden sich im Speicher an den Adressen $\delta + 0$ bis $\delta + 5$.

Aufgabe 19: (H) Multiprogramming

(11 Pkt.)

Beantworten Sie folgende Fragen zum Thema Multiprogramming:

- Was versteht man unter Multiprogramming?
- Was ist der Hauptvorteil von Multiprogramming?
- Nennen Sie jeweils einen Vor- und einen Nachteil eines Multiprozessor-Systems.

- d. Betrachten Sie nun die unten stehenden zwei Programme P und Q. A, B, C, D und E sind beliebige atomare (d.h. nicht unterbrechbare) Anweisungen. Die beiden Programme werden jeweils in einem eigenen Prozess pseudo-parallel ausgeführt:

```

1      PROGRAMM P;
2      BEGIN
3      A;
4      B;
5      C;
6      END;
```

```

1      PROGRAMM Q;
2      BEGIN
3      D;
4      E;
5      END;
```

- (i) Geben Sie alle möglichen Abläufe dieser Programme (in Form der Reihenfolgen der atomaren Anweisungen) an. Beispiel: Ein möglicher Ablauf wäre A B C D E (zuerst Prozess P komplett, dann Q komplett). Beachten Sie, dass die Anweisungsreihenfolge innerhalb eines Programms nicht verändert werden darf (z.B. darf B nicht vor A kommen).
- (ii) Nehmen Sie an, dass die Anweisung B Daten (z.B. Berechnungsergebnisse) erzeugt, die von der Anweisung E gelesen und weiterverarbeitet werden. Welches Problem kann sich aus dieser Abhängigkeit ergeben? Wann tritt es auf?

Aufgabe 20: (H) Einfachauswahlaufgabe: Prozesse

(5 Pkt.)

Für jede der folgenden Fragen ist eine korrekte Antwort auszuwählen („1 aus n“). Nennen Sie dazu in Ihrer Abgabe explizit die jeweils ausgewählte Antwortnummer ((i), (ii), (iii) oder (iv)). Eine korrekte Antwort ergibt jeweils einen Punkt. Mehrfache Antworten oder eine falsche Antwort werden mit 0 Punkten bewertet.

a) Wie bezeichnet man die Informationen, die den aktuellen Ausführungszustand eines Prozesses genau beschreiben (insb. die CPU-Register-Belegungen und alle Prozess-Status-Informationen)?						
(i) (Prozess-) Spiegel	(ii) (Prozess-) Quellcode	(iii) (Prozess-)Kontext	(iv) (Prozess-) Rahmen			
b) Wie bezeichnet man die sequenzielle, vollständige und unterbrechungsfreie Ausführung von Prozessen?						
(i) Uniprogramming	(ii) Multiprocessing	(iii) Multiprogramming	(iv) Broadprogramming			
c) Wie ist die mittlere Antwortzeit bei Prozessen mit folgender Ressourcennutzung unter Anwendung von Multiprogramming?						
Job	durchschnittliche CPU-Auslastung	Dauer	benötigter Speicher	Platte	Terminal	Drucker
1	50%	5 min.	50 KBytes	-	-	-
2	25%	20 min.	100 KBytes	-	ja	-
3	5%	15 min.	80 KBytes	ja	-	ja
(i) 20 min.		(ii) 23,3 min.	(iii) 5 min.	(iv) 13,3 min.		
d) Mit welchem Systemaufruf werden unter Unix/Linux Systemen neue Prozesse erzeugt?						
(i) creat	(ii) fork	(iii) execl	(iv) getpid			
e) Wie heißt der Prozess, der einen in Bearbeitung befindlichen Prozess unterbrechen und dem Prozessor einen anderen Prozess zuweisen kann?						
(i) Scheduler	(ii) Swapper	(iii) Dispatcher	(iv) Blocker			