

Betriebssysteme im Wintersemester 2008/2009

Übungsblatt 3

Abgabetermin: 29.10.2008, 13:30 Uhr

Aufgabe 9: (H) Nebenläufigkeit

(3 Pkt.)

Betrachten Sie das untenstehende Programm mit zwei Prozessen P und Q. A, B, C, D und E sind beliebige atomare (d.h. nicht unterbrechbare) Anweisungen. Die beiden Prozesse (realisiert als Prozeduren) werden parallel gestartet:

```
PROCEDURE P;  
2 BEGIN  
    A; B; C;  
4 END;  
  
6 PROCEDURE Q;  
    BEGIN  
8     D; E;  
    END;
```

- Geben Sie alle möglichen Abläufe dieses Programmes (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 einer Prozedur nicht verändert werden darf (z.B. darf B nicht vor A kommen).
- 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 10: (H) Zustands-Prozessmodelle

(11 Pkt.)

- a. Geben Sie für jeden der folgenden Zustandsübergänge an, ob der Übergang zulässig ist und auf welche Weise der Übergang stattfindet oder warum kein solcher Übergang möglich ist:
 - (i) Ändern des Prozesszustandes von „Blocked“ zu „Running“.
 - (ii) Ändern des Prozesszustandes von „Running“ zu „Blocked“.
 - (iii) Ändern des Prozesszustandes von „Ready“ zu „Blocked“.
- b. Ausgehend vom 5-Zustands-Prozessmodell:
 - (i) Geben Sie für jeden Zustandsübergang ein Beispiel an, das den jeweiligen Zustandswechsel auslösen könnte.
 - (ii) Welche Zustände kann man einsparen, wenn das Betriebssystem in reinem Batch-Betrieb (kein Multiprogramming) arbeitet?
 - (iii) Wie ändert sich das Modell für ein Zwei-Prozessor-System?
- c. Ausgehend vom 7-Zustands-Prozessmodell: Geben Sie für jeden Zustandsübergang in einen Suspendiert-Zustand ein Beispiel an, das den jeweiligen Zustandswechsel auslösen könnte.
- d. Sind die folgenden Aussagen wahr oder falsch?
 - (i) Mögliche Prozesszustandsübergänge im 2-Zustands-Prozessmodell sind „enter“, „dispatch“, „run“ und „schedule“.
 - (ii) Im 7-Zustands-Prozessmodell spricht man von einem Timeout, falls ein Prozess auf das Eintreten eines bestimmten Ereignisses warten muss.
 - (iii) Im 7-Zustands-Prozessmodell gibt es einen Übergang von „Running“ nach „Ready“.
 - (iv) Im 7-Zustands-Prozessmodell gibt es einen Übergang von „Running“ nach „Blocked“.
 - (v) Der Prozessdeskriptor ist Teil des Prozesskontrollblocks (PCB).

Aufgabe 11: (K) Prozesszustände

(7+2+1 Pkt.)

- a. Skizzieren Sie das 5-Zustands-Prozessmodell und erklären Sie jeweils kurz die einzelnen Zustände sowie die Übergänge.
- b. Diskutieren Sie, wie sich das Hinzufügen von Prozesszuständen in ein Prozessmodell auf die *Prozessverwaltung* auswirkt.
- c. Erklären Sie kurz die Begriffe *Scheduling* und *Dispatching*.

Aufgabe 12: (T) Kontext- und Moduswechsel

(9 Pkt.)

- Was versteht man unter einem Moduswechsel? Wie läuft er ab?
- Welche Aktionen muss das Betriebssystem bei einem Kontextswitch zwischen Prozessen vornehmen?
- Wovon hängt der Aufwand für einen Kontextswitch also im Wesentlichen ab?
- Geben Sie jeweils ein Beispiel für eine Situation an, die
 - nur einen Moduswechsel,
 - einen Kontextswitch (je nach Architektur mit implizitem Moduswechsel, um die Process-Switching-Routine des Betriebssystems aufzurufen),
 - einen Moduswechsel mit anschließendem Kontextswitch erfordert.
- Sie werden von der Firma AB Computer angestellt, um die Geschwindigkeit ihrer Systeme zu verbessern. Ihre Anwendungen nutzen nur 10 der 32 Register der CPU; daher wird vorgeschlagen, die Kontextswitching-Routine des Betriebssystems so zu verändern, dass nur die zehn benötigten Register gesichert werden. Nehmen Sie an, dass Sie die Kontextswitching-Routine korrekt ändern können. Ist dies eine gute oder eher schlechte Idee? Warum?

Aufgabe 13: (T) Prozesszustände I

(10 Pkt.)

Die folgende Tabelle stellt die Prozeßzustände des VAX/VMS-Betriebssystems dar:

Prozeßzustand	Beschreibung
Currently executing, CUR	Prozeß wird ausgeführt (<i>Running</i>)
Computable (resident), COM	Bereit und im Hauptspeicher
Computable (outswapped), COMO	Bereit, aber ausgelagert
Page fault wait, PFW	Prozeß hat auf einen Speicherbereich zugegriffen, der sich nicht im Hauptspeicher befindet und muß darauf warten, bis dieser Bereich geladen ist.
Collided page wait, COLPG	Prozeß hat auf einen Speicherbereich zugegriffen, der gerade aus- oder eingelagert wird.
Common event flag wait (resident), CEF	Wartend auf ein shared event flag (i.A. für Interprozesskommunikation verwendet).
Common event flag wait (outswapped), CEFO	analog, ausgelagert
Free page wait, FPG	Wartend bis ein Speicherbereich zu dem aktuellen Speicherbereich des Prozesses hinzugefügt wird.
Hibernate wait (resident), HIB	Prozess versetzt sich selbst in Ruhezustand.
Hibernate wait (outswapped), HIBO	analog, ausgelagert.
Local event flag wait (resident), LEF	Prozess wartet auf ein lokales Ereignis (i.a. E/A-Operation), im Speicher.
Local event flag wait (outswapped), LEFO	analog, ausgelagert.
Suspended wait (resident), SUSP	Prozeß durch anderen Prozeß in Wartezustand.
Suspended wait (outsw.), SUSPO	analog, ausgelagert.
Mutex/Miscellaneous wait, MWAIT	Prozeß wartet auf verschiedene Systemressourcen.

- Überlegen Sie sich eine Begründung für die Existenz derartig vieler Prozeßzustände!
- Warum haben einige Zustände keine Unterscheidung in resident und outswapped?
- Zeichnen Sie das Zustandsübergangsdiagramm und benennen Sie für jeden Zustandsübergang die Aktion oder das Ereignis, das ihn bewirkt.