Abschlussklausur

Betriebssysteme (BTS)

6. Juli 2007

Name:
Vorname:
Matrikelnummer:
Studiengang:

Hinweise:

- Tragen Sie zuerst auf allen Blättern (einschließlich des Deckblattes) Ihren Namen, Ihren Vornamen und Ihre Matrikelnummer ein. Lösungen ohne diese Angaben können nicht gewertet werden.
- Schreiben Sie die Lösungen jeder *Teil*aufgabe auf das jeweils vorbereitete Blatt. Erst wenn Vorder- und Rückseite nicht ausreichen, können die leeren Blätter am Ende der Heftung benutzt werden. In diesem Fall ist ein Verweis notwendig. Eigenes Papier darf nicht verwendet werden.
- Legen Sie bitte Ihren Lichtbildausweis bereit.
- Als *Hilfsmittel* sind zwei selbstständig beschriebene DIN-A4-Blätter und Taschenrechner zugelassen.
- Mit Bleistift oder Rotstift geschriebene Ergebnisse werden nicht gewertet.
- Die Bearbeitungszeit für die Klausur beträgt 70 Minuten.
- Stellen Sie sicher, dass Ihr Mobiltelefon ausgeschaltet ist. Klingelnde Mobiltelefone werden als Täuschungsversuch angesehen und der/die entsprechende Student/in wird von der weiteren Teilnahme an der Klausur ausgeschlossen!

Bewertung:

1a)	1b)	2a)	2b)	2c)	3)	4)	5a)	5b)	5c)	5d)
6)	7a)	7b)	7c)	8a)	8b)	Σ	Note			

Abschlussklausur

Betriebssysteme (BTS)

6.7.2007 MSc Christian Baun

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

- a) Nennen Sie 5 **Ersetzungsstrategien** aus der Cache-Datenverwaltung. Es ist nur eine Aufzählung und keine Beschreibung der Funktionalität verlangt!
- b) Es existieren zwei Möglichkeiten für Schreibzugriffe auf den Cache.
 - Nennen Sie die beiden Cache-Schreibstrategien.
 - Beschreiben Sie kurz (1-2 Sätze) die Funktionsweise der beiden Cache-Schreibstrategien.
 - Nennen Sie für jede der beiden Cache-Schreibstrategien jeweils eine vorteilige und eine nachteilige Eigenschaft.

Aufgabe 2 (2+1+3 Punkte)

- a) Das **2-Zustands-Prozessmodell** ist das kleinste, denkbare Prozessmodell. Zeichnen Sie das 2-Zustands-Prozessmodell mit seinen Zuständen und allen Prozessübergängen.
- b) Ist das 2-Zustands-Prozessmodell sinnvoll? Begründen Sie kurz ihre Antwort.
- c) Zeichnen Sie das 5-Zustands-Prozessmodell mit seinen Zuständen und allen Prozessübergängen.

Aufgabe 3 (4 Punkte)

Beschreiben Sie die Funktionsweise der beiden UNIX-Systemaufrufe fork() und exec(). Heben Sie dabei die Unterschiede zwischen beiden Systemaufrufen hervor.

Aufgabe 4 (2 Punkte)

Die existierenden **Schedulingverfahren** können in zwei grundsätzliche Klassen unterteilt werden. Welche sind das und wie unterscheiden sich diese?

Aufgabe 5 (2,5+2,5+2,5+2,5 Punkte)

Auf einem Einprozessorrechner sollen fünf Prozesse verarbeitet werden. Hohe Prioritäten sind durch hohe Zahlen gekennzeichnet.

Prozess	CPU-Laufzeit (ms)	Priorität
1	4	5
2	8	12
3	4	8
4	8	2
5	6	10
6	4	4

- a) Skizzieren Sie die Ausführungsreihenfolge der Prozesse mit einem Gantt-Diagramm (Zeitleiste) für Round Robin (Zeitquantum q = 2 ms), FCFS, SJF, LJF und Prioritätengesteuertes Scheduling.
- b) Berechnen Sie die mittleren Laufzeiten der Prozesse.
- c) Berechnen Sie die mittleren Wartezeiten der Prozesse.
- d) Welche der obigen Scheduling-Verfahren können zu einem Verhungern der Prozesse führen und welche nicht?

Aufgabe 6 (15 Punkte)

In einer Lagerhalle der John Doe Werke werden ständig Pakete von einem Lieferanten angeliefert und von zwei Auslieferern abgeholt. Der Lieferant und die Auslieferer müssen dafür ein Tor durchfahren. Das Tor kann immer nur von einer Person durchfahren werden. Der Lieferant bringt mit jeder Lieferung 3 Pakete zum Wareneingang, an der Ausgabe holt ein Auslieferer jeweils 2 Pakete ab, der andere Auslieferer 1 Paket. Die Prozesse laufen folgendermassen ab:

```
Lieferant
                                      Auslieferer_X
                                                                           Auslieferer Y
{
                                      {
                                                                           {
  while (TRUE)
                                        while (TRUE)
                                                                             while (TRUE)
  {
                                        {
                                                                              {
    <Tor durchfahren>;
                                          <Tor durchfahren>;
                                                                                <Tor durchfahren>;
    <Wareneingang betreten>;
                                          <Warenausgabe betreten>;
                                                                                <Warenausgabe betreten>;
    <3 Pakete entladen>;
                                          <2 Pakete aufladen>;
                                                                                <1 Paket aufladen>;
                                          <Warenausgabe verlassen>;
    <Wareneingang verlassen>;
                                                                                <Warenausgabe verlassen>;
    <Tor durchfahren>;
                                          <Tor durchfahren>;
                                                                                <Tor durchfahren>;
}
                                     }
                                                                           }
```

Es existiert genau ein Prozess Lieferant, ein Prozess Auslieferer_X und ein Prozess Auslieferer_Y.

Synchronisieren Sie diese drei Prozesse, indem Sie die notwendigen **Semaphoren** deklarieren, diese mit Startwerten versehen und Semaphor-Operationen einfügen. Folgende Bedingungen müssen erfüllt sein:

- Es darf immer nur ein Prozess das Tor durchfahren.
- Es darf immer nur einer der beiden Auslieferer die Warenausgabe betreten. Es soll aber zugelassen werden, dass der Lieferant und ein Auslieferer gleichzeitig Waren entladen bzw. aufladen.
- Die Lagerhalle hat nur eine beschränkte Kapazität. es können maximal 10 Pakete aufgenommen werden.
- Es dürfen keine Verklemmungen auftreten, da sonst die John Doe Werke viel Zeit und Geld verlieren.
- Zu Beginn sind keinerlei Pakete in der Lagerhalle vorrätig und das Tor, der Wareneingang und die Warenausgabe sind frei.

Aufgabe 7 (1+1+1 Punkte)

- a) Was ist die Aufgabe eines **Mutex**?
- b) Welche Zustände kann ein Mutex annehmen?
- c) Was sind die Unterschiede zwischen Mutexen und Semaphoren?

Aufgabe 8 (2+4 Punkte)

- a) Welche Auswirkungen hat die Größe der Cluster im Dateisystem?
- b) Was sind **Journaling-Dateisysteme**? Wie ist die Funktionsweise von Journaling-Dateisystemen? Was sind die Vorteile von Journaling-Dateisystemen gegenüber Dateisystemen ohne Journal?

Name:	Vorname:	Matr.Nr.:
Aufgabe	1)	Punkte:

Name:	Vorname:	Matr.Nr.:
Aufgabe	2)	Punkte:

Name:	Vorname:	Matr.Nr.:
Aufgabe 3)		Punkte:

Name:	Vorname:	Matr.Nr.:
Aufgabe 4)	Punkte:

Name:	Vorname:	Matr.Nr.:
Aufgabe 5)		Punkte:

Name:	Vorname:	Matr.Nr.:
Aufgabe	6)	Punkte:

Name:	Vorname:	Matr.Nr.:
Aufgabe 7)		Punkte:

Name:	Vorname:	Matr.Nr.:
Aufgabe 8)		Punkte:

Zusatzblatt zu Aufgabe.....

Verwenden Sie dieses Blatt nur für eine Teilaufgabe! Verweisen Sie bei der zugehörigen Aufgabe gut sichtbar auf dieses Blatt!

Zusatzblatt zu Aufgabe.....

Verwenden Sie dieses Blatt nur für eine Teilaufgabe! Verweisen Sie bei der zugehörigen Aufgabe gut sichtbar auf dieses Blatt!

Zusatzblatt zu Aufgabe.....

Verwenden Sie dieses Blatt nur für eine Teilaufgabe! Verweisen Sie bei der zugehörigen Aufgabe gut sichtbar auf dieses Blatt!