### Abschlussklausur

### Betriebssysteme (BTS)

7. Juli 2011

Name:
Vorname:
Matrikelnummer:
Studiengang:
Tinwoise:

- 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 Teilaufgabe auf das jeweils vorbereitete Blatt. Sie können auch die leeren Blätter am Ende der Heftung nutzen. In diesem Fall ist ein Verweis notwendig. Eigenes Papier darf nicht verwendet werden.
- $\bullet$  Legen Sie bitte Ihren Lichtbildausweis und Ihren Studentenausweis bereit.
- Als *Hilfsmittel* ist ein Taschenrechner zugelassen.
- Mit Bleistift oder Rotstift geschriebene Ergebnisse werden nicht gewertet.
- Die Bearbeitungszeit dieses Teils der Abschlussklausur beträgt 60 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:

1)	2)	3)	4)	5)	6)	7)	8)	9)	Σ	Note

#### Abschlussklausur

### Betriebssysteme (BTS)

7.7.2011 M.Sc. Christian Baun

#### Aufgabe 1 (6+2 Punkte)

- a) Der Speicher eines Computersystems wird in die drei Kategorien Primärspeicher, Sekundärspeicher und Tertiärspeicher unterschieden. Beschreiben Sie die qualitativen Merkmale dieser Speichersorten und ihre Ausprägungen. Zeichnen Sie dazu ein Diagramm. Nennen Sie auch mindestens zwei Beispiele pro Kategorie.
- b) Der **Tertiärspeicher** wird ebenfalls in zwei Kategorien unterschieden. Benennen Sie diese beiden Kategorien und beschreiben Sie diese.

#### Aufgabe 2 (1+2 Punkte)

- a) Es existieren zwei grundsätzliche Konzepte, um **Schreibzugriffe auf Cache** durchzuführen. Welche beiden Konzepte sind das?
- b) Beschreiben Sie die beiden Konzepte. Gehen Sie auf die Unterschiede, Vor- und Nachteile ein.

#### Aufgabe 3 (5 Punkte)

Zeichnen Sie das 5-Zustands-Prozessmodell mit seinen Zuständen und allen Prozessübergängen.

### Aufgabe 4 (3 Punkte)

Nennen Sie die drei Arten von **Prozesskontextinformation**, die das Betriebssystem speichert und beschreiben Sie deren Inhalt.

#### Aufgabe 5 (2+2 Punkte)

- a) Was ist ein **Dispatcher** und was sind seine Aufgaben?
- b) Was ist ein **Scheduler** und was sind seine Aufgaben?

### Aufgabe 6 (4+1+1+1 Punkte)

- a) Beschreiben Sie den Ablauf der verbindungsorientierten Kommunikation mit Sockets. Gehen Sie dabei besonders auf die nötigen Operationen ein. Zeichnen Sie dazu ein Diagramm.
- b) Bei welcher Art der Interprozesskommunikation müssen die beteiligten Prozesse verwandt sein?
- c) Bei welcher Art der Interprozesskommunikation müssen die Entwickler sich selbst um die **Synchronisierung der Schreibzugriffe** kümmern?
- d) Bei welchen **Pipes** kann die Interprozesskommunikation **bidirektional** erfolgen und bei welchen Pipes kann die Interprozesskommunikation nur **unidirektional** erfolgen?

#### Aufgabe 7 (6+6+6 Punkte)

Auf einem Einprozessorrechner sollen sieben Prozesse verarbeitet werden.

Prozess	CPU-Laufzeit (ms)	Ankunftszeit (ms)
A	5	0
В	7	3
С	2	5
D	6	7
E	1	10
F	5	18
G	4	24

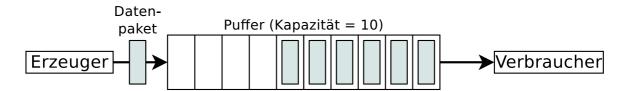
- a) Skizzieren Sie die Ausführungsreihenfolge der Prozesse mit einem Gantt-Diagramm (Zeitleiste) für Round Robin (Zeitquantum q=1 ms), Longest Remaining Time First (LRTF) und Shortest Remaining Time First (SRTF).
- b) Berechnen Sie die mittleren Laufzeiten der Prozesse.
- c) Berechnen Sie die mittleren Wartezeiten der Prozesse.

#### Aufgabe 8 (2+2+1 Punkte)

- a) Was ist eine Semaphore und was ist ihr Einsatzzweck?
- b) Welche beiden **Operationen** werden bei Semaphoren verwendet? Gesucht sind die Bezeichnungen und eine (kurze) Beschreibung der Funktionsweise.
- c) Was ist der Unterschied zwischen Semaphoren und Sperren?

#### Aufgabe 9 (7 Punkte)

- Erzeuger/Verbraucher-Szenario.
  - Ein Erzeuger soll Daten an einen Verbraucher schicken
  - Ein endlicher Zwischenspeicher (Puffer) soll die Wartezeiten des Verbrauchers minimieren
  - Daten werden vom Erzeuger in den Puffer gelegt und vom Verbraucher aus diesem entfernt
  - Gegenseitiger Ausschluss ist notwendig, um Inkonsistenzen zu vermeiden
  - Ist der Puffer voll, muss der Erzeuger blockieren
  - Ist der Puffer leer, muss der Verbraucher blockieren



• Aufgabe: Synchronisieren Sie die beiden Prozesse, indem Sie die notwendigen Semaphoren deklarieren, diese mit Startwerten versehen und Semaphor-Operationen einfügen.

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	5)	Punkte:

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

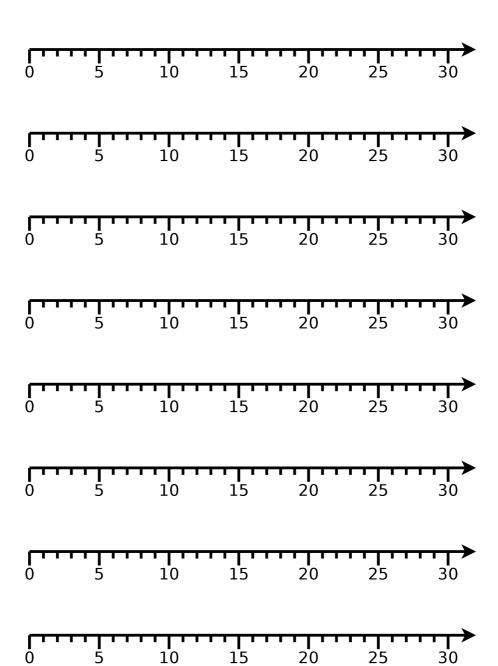
Name:

Vorname:

Matr.Nr.:

## Aufgabe 7a)

Punkte: .....



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

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

Name:	Vorname:	Matr.Nr.:
	,	
Aufgabe	2 8)	Punkte:

Name: Vorname: Matr.Nr.:

# Aufgabe 9)

```
typedef int semaphore;
void erzeuger (void) {
  int daten;
    ile (TRUE) { // Endlosschleife erzeugeDatenpaket(daten); // erzeuge Datenpaket
  while (TRUE) {
    einfuegenDatenpaket(daten); // Datenpaket in Puffer schreiben
 }
void verbraucher (void) {
  int daten;
  while (TRUE) {
                                  // Endlosschleife
    entferneDatenpaket(daten); // Datenpaket aus Puffer holen
    verbraucheDatenpaket (daten); // Datenpaket nutzen
 }
}
```

# 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!