



- Probieren Sie ein paar der vorgefertigten *Themens* aus, um das Aussehen der Folien zu ändern
- Fügen Sie ein beliebiges Bild in eine Folie ein.

Freiburg, 6. Dezember 2007

## Systeme I - WS 07/08 Übungsblatt 4 - Praxis L<sup>A</sup>T<sub>E</sub>X (Forts.)

### Aufgabe 1

Wenn Sie die Aufgaben vom letzten Mal nicht vollständig gelöst haben, lösen Sie zuerst die dortigen Aufgaben.

### Aufgabe 2

Legen Sie eine einfache Vorlage für eine `Latex`-Eingabedatei an. Benutzen Sie die Vorlage, um einen beliebigen Text (länger als eine Seite) in `LATEX` umzusetzen.

- Erzeugen Sie ein Titelblatt
- Machen Sie einen Querverweis von einem Teil eines Textes zu einem anderen Teil
- Heben Sie ein *Textstück* hervor. Markieren Sie ein Textstück **Pett**
- Probieren Sie verschiedene Schriftgrade und Schriftstile aus
- Erzeugen Sie eine Fußnote
- Machen Sie den Text zweispaltig.

### Aufgabe 3

Erstellen Sie eine `Latex`-Eingabedatei, die die Tabelle aus der Aufgabe 1 des Theorie-Teils (Übungsblatt 4) enthält.

### Aufgabe 4

Die *Beamer* `LATEX`-Klasse (<http://latex-beamer.sourceforge.net/>) wird benutzt um Präsentationen zu erstellen. Dabei können die Folien durchaus graphisch ansprechend mit verlinkten Navigationsstrukturen, Farbübergängen oder Videos gestaltet sein.

- Machen Sie sich mit der `Beamer`-Klasse vertraut und erstellen Sie mit Hilfe dieser Klasse verschiedene Folien

**Abgabe: Vorführung nächste Woche in den Übungsgruppen.**



Freiburg, 6. Dezember 2007

**Systeme I - WS 07/08**  
**Übungsblatt 4 - Theorie**

**Aufgabe 1:**

Drei Prozesse ( $p_1$ ,  $p_2$ ,  $p_3$ ) greifen auf drei Ressourcen (R, S, T) zu. Zeigen Sie, daß es eine Abarbeitungsfolge der folgenden Prozesse gibt, so daß ein Deadlock auftritt:

#	Prozess 1	Prozess 2	Prozess 3
1	Anforderung R	Anforderung S	Anforderung R
2	Anforderung S	Anforderung T	Anforderung S
3	Freigabe R	Freigabe S	Anforderung T
4	Anforderung T	Anforderung R	Freigabe T
5	Freigabe S	Freigabe R	Freigabe R
6	Freigabe T	Freigabe T	Freigabe S

Benutzen Sie einen Belegungs-Anforderungs-Graphen, wie er in der Vorlesung vorgestellt wurde; geben Sie die einzelnen Zwischenschritte des Graphen an.

**Aufgabe 2:**

Unter bestimmten Umständen es kann bei der von Tanenbaum vorgeschlagenen Lösung des *Dining Philosophers Problem* (siehe Vorlesungsfolien) allerdings dazu kommen, dass ein Philosoph verhungert. Zeigen Sie dies mit einem Gegenbeispiel (bitte gut dokumentieren).

Hinweis:

Gehen Sie von 5 gefräßigen Philosophen aus, d.h. sie denken nur kurz und wollen gleich wieder essen. Beginnen Sie mit einer Konfiguration, bei der zwei Philosophen gerade essen und die anderen drei hungrig und blockiert sind.

**Aufgabe 3:**

Zwei Prozesse P0 und P1 wollen auf eine Ressourcenklasse A zugreifen. Folgende Tabelle zeigt, in welcher Reihenfolge die Prozesse Ressourcen anfragen bzw. freigeben. Die Anzahl der zur Verfügung stehenden Ressourcen V sei 2. Die Variable x wird von den Prozessen gemeinsam genutzt.

**Prozess 0**

- 1:  $x = -1$ ;
- 2: Anforderung A;
- 3: **if** ( $x >= 0$ ) { Freigabe A; }
- 4: Anforderung A;
- 5: Freigabe A;
- 6: **if** ( $x < 0$ ) { Freigabe A; }

**Prozess 1**

- 1: Anforderung A;
- 2:  $x = 5$ ;
- 3: Anforderung A;
- 4: Freigabe A;
- 5: Freigabe A;

(a) Bestimmen Sie für jeden Prozess die maximale Anzahl  $M_i$  von Ressourcen. Sind die Voraussetzungen für den Bankieralgorithmus erfüllt? Begründen Sie kurz Ihre Aussage.

(b) Sei die Ausführung jeder Zeile eine atomare Operation. Betrachten Sie folgenden Zustand: P0 führt zuerst Zeile 1 und 2 aus, danach führt P1 die Zeilen 1 und 2 aus. Warum ist dieser Zustand bezüglich des Bankieralgorithmus unsicher?

(c) Gibt es ausgehend von dem in Teil b) angegebenen unsicheren Zustand eine Abarbeitungsreihenfolge der Prozesse, die zu einem Deadlock führt?

Abgabe: Freitag, 21.12.2007 vor der Vorlesung.