

Name, Matrikelnummer

<b>Prüfer:</b>	Prof. Dr.-Ing. Rainer Keller	<b>Anzahl der Seiten:</b>	9
<b>Studiengänge:</b>	Softwaretechnik und Medieninformatik Technische Informatik Ingenieurpädagogik	<b>Semester:</b>	SWB2 TIB2 IEP2
<b>Klausur:</b>	<b>Betriebssysteme</b>	<b>Prüfungsnummern:</b>	IT 105 2004
<b>Hilfsmittel:</b>	keine, außer 1 DIN A4 Blatt, beidseitig von Hand selbst beschrieben	<b>Dauer der Klausur:</b>	90 Minuten

Bitte lesen Sie die Aufgaben sorgfältig durch. Jede Aufgabe besteht aus Unteraufgaben – für die es Teilpunkte gibt. Jeder Punkt entspricht ca. 1 Minute Arbeitszeit. Nutzen Sie also den zur Verfügung stehenden Raum und die Zeit aus, um möglichst sorgfältig und ausführlich zu antworten.

## Allgemeines

**(9 Punkte)**

a) Was war so exorbitant an der Entwicklung der Mikroprozessoren?

• Die Entwicklung von Intel x86, AMD, ARM war exorbitant, da:

- ~ 1980: integrated cache
- ~ 1984: Multiple instructions per cycle
- ~ 2006: Multi-Core → mit mehreren Kernen umgehen
- ~ 2016: 40 cores (Spectre & Meltdown → man muss erst die tiny-Schwachstelle sichern, um BS sicher zu machen)

Mooresches Gesetz

3

b) Wie lautet die in der Vorlesung gegebenen Definition eines Betriebssystems?

• Ein BS ist eine SW, die auf effiziente Weise die Komplexität eines Computers vor dem User und dem Programmierer versteckt und eine Gruppe von Usern und Programmen gemeinsamen, sicheren Zugriff auf Rechen- (CPU, GPU), Speicher- (RAM, Festplatte, CD/DVD), Kommunikationsmittel (Netzwerkadapter, GPU, Tastatur) zur Verfügung stellt.

4

c) Circa wie groß ist ein L3 Cache eines modernen Intel Prozessors?

L3 Cache ist wichtig & riesig ca. 20MB / 50MB

2

\_\_\_\_\_  
Name, Matrikelnummer

## Bash Shell

(17 Punkte)

a) Benennen Sie für die jeweilige Shell-Funktion den passenden Shell-Befehl:

Größe aller Dateien im Verzeichnis:	<code>du -hs</code>
Hilfeseite von bash aufrufen:	<code>man -k</code>
Prozess 666 hart beenden:	<code>kill -9 666</code>
Top Prozesse auflisten:	<code>top</code>
Datum ausgeben:	<code>date</code>
Archive ver-/auspacken:	<code>tar</code>
Prozess in den Hintergrund:	<code>bg</code>
Zeilen zählen:	<code>wc</code>

4

b) Die Umgebungsvariable `DATEI` enthält den Namen einer Datei. Aus dieser Datei sollen Sie Vorname, Name, Matrikelnummer und Note von Studenten auslesen. Die Daten sind jeweils durch Doppelpunkte getrennt, die Note ist zweistellig, bspw.

`Max:Meier:12345:20`

`Moritz:Mueller:23456:1`

`Miriam:Musterfrau:34567:10`

Schreiben Sie ein Shell-Script, welches die Matrikelnummer des Studenten mit der besten Note ausgibt.

7

Name, Matrikelnummer

c) Was sind systemnahe Programme? Bitte geben Sie Beispiele an.

Systemnahe Programme in BS sind Anw., die direkt mit der HW und den unteren Ebenen des BS interagieren.  
Bsp: kernel  
Systemcalls  
Shell

3

d) Was machen die beiden folgenden Programme?

mount	Bindet ein Dateisystem im Verzeichnisbaum ein.
fsck	Dateisystem, welches auf Device befindet, wird überprüft.
mknod	Auf Geräte <sup>wird</sup> mittels spez. Dateien identifiziert → wenn man selber anlegen

3

Name, Matrikelnummer

**Scheduling und Systemcalls****(22 Punkte)**

- a) Nennen Sie drei Optionen zum Setzen der Häufigkeit der Unterbrechung der CPU durch den Linux Scheduler – und was sind deren Vor- / Nachteile

Wie häufig?	Vor- / Nachteil
1.  100 Hz.	Desktop System: guter Kompromiss für viele Anw.
2.  1000 Hz	Server System: mit vielen 1000 Prozessen (spez. Systeme zur Verarbeitung von Audiodateien) (Netzwerkzugriffe)
3.  Variabel	Laptop / System: zur numerischen Datenverarbeitung (wenige/eine Anw.)

5

- b) Welche beiden (temporären) Informationen verarbeitet ein CPU-Scheduler, um die Priorität eines Tasks neu zu berechnen? Wieso priorisiert er bestimmte Tasks höher und welche Tasks sind das?

<p>1.) Rechenintensiver Prozess (Webserver)</p> <p>2.) I/O intensiver Prozess (Web Browser)</p> <p>Linux Scheduler enthält Aspekte aus versch. Verfahren</p> <ul style="list-style-type: none"> <li>• OS unterbricht (pre emptiv) alle laufenden Anw. und teilt CPU zu.</li> <li>• Damit OS unterscheiden kann, welcher Prozess auszuführen ist, werden Parameter übermittelt.</li> </ul> <p>Tasks:</p> <ul style="list-style-type: none"> <li>• First in First Out (FIFO)</li> <li>• First Come First Served (FCFS)</li> <li>• Shortest Job First</li> <li>• Round Robin</li> </ul>	3
--	---

- c) Wie kann man Systemcalls einer Anwendung mit der PID 123 mitlesen?

<p>Strace -p 123</p>	2
----------------------	---

Name, Matrikelnummer \_\_\_\_\_

d) Beurteilen Sie jede Aussage ob sie Wahr (W) oder Falsch (F) ist:

Aussage	W/F?
1:1 Modell heißt ein User-Thread ist ein Linux-Task	W
Das 1:1 Modell reduziert Komplexität	W
Früher hatte Linux ein 1:n Task:Thread Modell	F
Zur besseren Lastbalance werden Tasks auf freie Cores verschoben	W
Die <code>vruntime</code> ist in Millisekunden (ms) aufgeteilt	F
In <code>struct task{}</code> stehen die lauffähigen und gestoppten Tasks	F
Ein Task läuft nachdem er auf einem core gestartet wurde nur auf diesem	F
<small>Es gibt nur einen Thread</small> Ein Prozess besteht aus einem Thread und ist ein Task	F
Die CPU-Zeit wird an die <code>nr_running</code> Tasks verteilt	W

run key =>  
multiple tasks

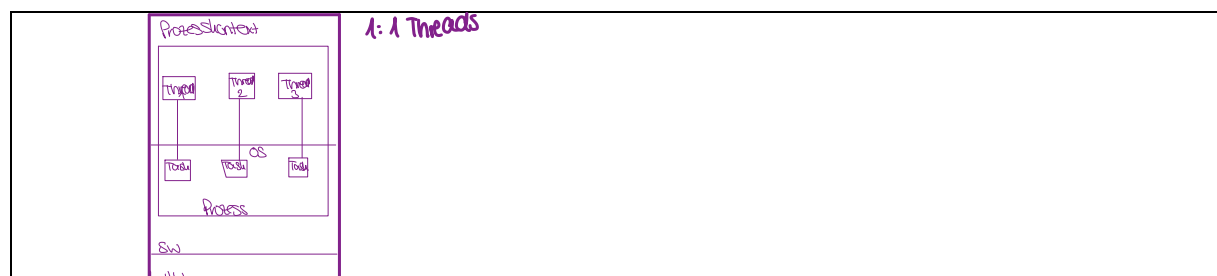
5

e) Welche Ressourcen teilen sich alle Threads eines Prozesses, welche sind pro Thread?

<b>Geteilte Ressourcen?</b> <ul style="list-style-type: none"> <li>• Open File Descriptor (FDs) <small>-einen Sockets</small></li> <li>• Speicher → alles was alloziert</li> <li>• Signal Handler</li> </ul>	
<b>Per-Thread Ressourcen?</b> <ul style="list-style-type: none"> <li>• Registersatz</li> <li>• Stack</li> <li>• Thread-ID, Priority</li> <li>• Fehlervariable <code>errno</code></li> </ul>	

3

f) Malen Sie die Zuordnung von Linux Threads eines Prozesses auf Tasks, sodaß die Zugehörigkeit zu den Ebenen klar wird:



4

Name, Matrikelnummer

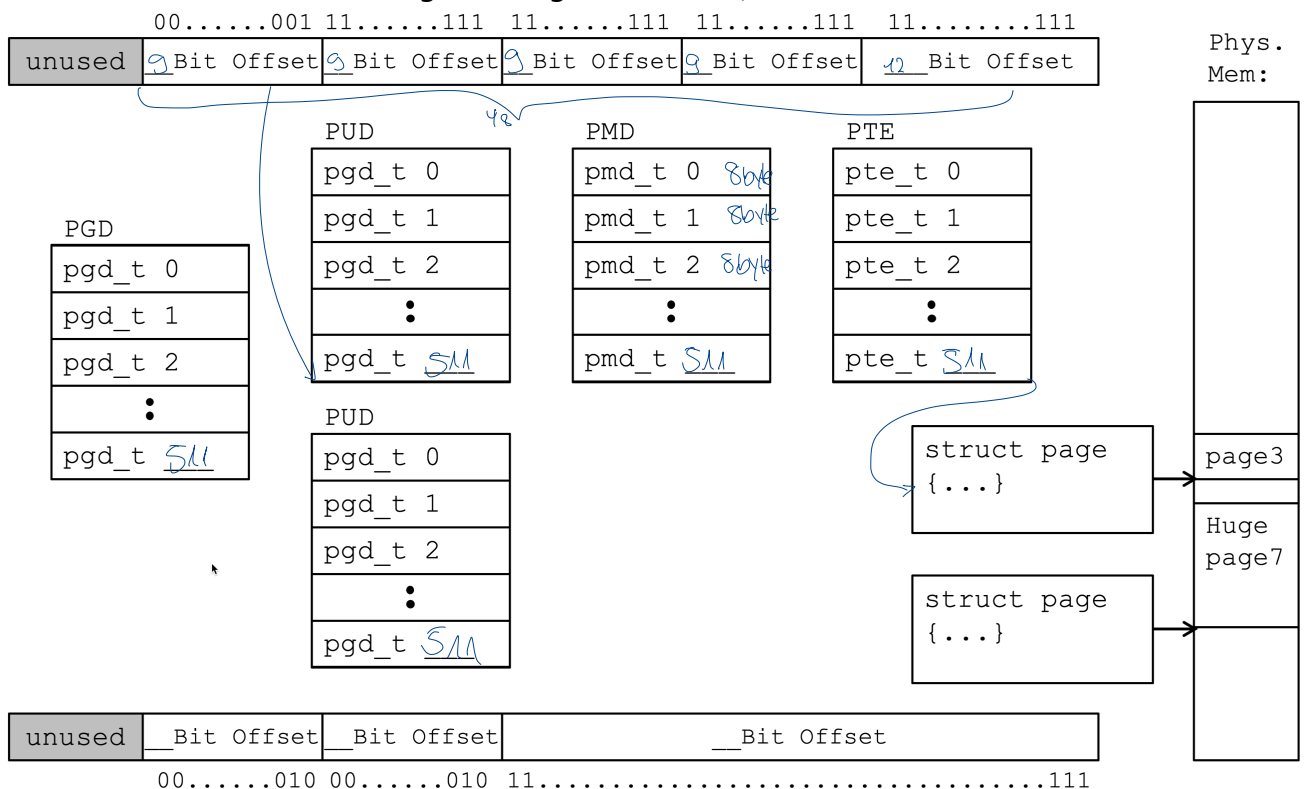
## Virtueller Speicher

(19 Punkte)

- a) Zeichnen Sie die Indexe und Pointer ein für die folgenden 64-Bit Virtuellen Adressen – bitte **beachten** Sie die binären Zahlenwerte (0...001 bedeutet eine 1 im niederwertigsten Bit, ansonsten Nullen). Geben Sie weiterhin die Anzahl Bits an den **unterstrichenen** Stellen ein.

8

Sollten Sie Offset-Längen korrigieren wollen, nutzen Sie bitte das Feld unten.



Korrektur:

\_\_\_\_\_  
Name, Matrikelnummer

- b) Der Data Segment Descriptor bietet 3 Bits für Speicherschutz: Writable, Accessed und Present. Welche Konzepte lassen sich mit diesen Bits umsetzen?

4

- c) Der Intel Prozessor unterstützt 4 Ebenen für den Priviledge Level (CPL). Welche Softwareschicht auf einem Betriebssystem beispielhaft zugeordnet sein?

0. Ebene:	
1. Ebene:	
2. Ebene:	
3. Ebene:	

4

- d) Der Buddy Allocator für Speicherverwaltung hat ein Problem mit Fragmentierung. Wieso ist das für Verwaltung von VM-Pages kein Problem?

3

\_\_\_\_\_  
Name, Matrikelnummer

## Interprozesskommunikation

(14 Punkte)

a) Bitte geben Sie min. einen Unix-Funktionsaufruf je Kommunikationsmodell an:

Direkt in Speicher eines Prozesses schreiben:	
Einem Prozess eine Benachrichtigung schicken:	
Datei beschreiben:	
Uni-direktionale Verbindung:	

2

b) Warum könnte eine Uni-direktionale Verbindung langsamer sein – aber wieso ist das Konzept dennoch so gut?

Gründe für langsame Performance:
Gründe für gutes Konzept:

6

c) Nennen Sie den vermutlich schnellsten IPC-Kommunikation bezüglich Bandbreite und eine Begründung warum:

Schnellster Call:	Begründung:

6



---

 Name, Matrikelnummer

## Dateisysteme

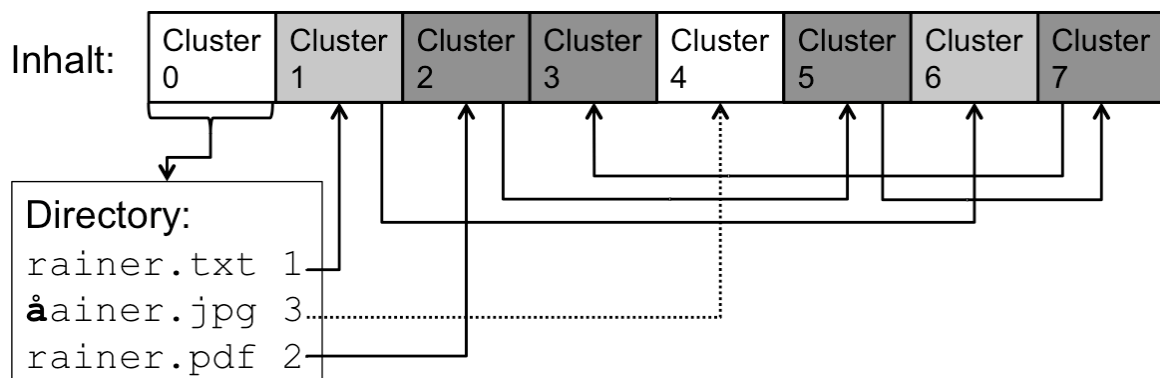
**(9 Punkte)**

- a) Zur korrekten Unterscheidung von Datenträgergrößen wurde eine IEC-Norm eingefügt. Nennen Sie mindestens 3 Größen und geben grob an was das an Unterschied ausmacht.

Bezeichnung	Prozentualer Unterschied

3

- b) Defragmentieren Sie das unten stehende FAT-Verzeichnis. Geben Sie Schritte mit an, sowie die Folge der Cluster je Datei.



1.	2.	3.
4.	5.	6.
Resultat für Folge rainer.txt		
Resultat für Folge rainer.pdf:		

6