Name, Matrikelnummer

Prüfer:	Prof. Dr.-Ing. Rainer Keller	**Anzahl der Seiten:**	14
Studiengänge:	Softwaretechnik und Medieninformatik Technische Informatik Ingenieurpädagogik	**Semester:**	SWB2 TIB2 IEP2
Klausur:	**Betriebssysteme**	**Prüfungsnummern:**	IT 105 2004
Hilfsmittel: **keine**, außer **1 DIN A4** Blatt, beidseitig **selbst** geschrieben (keine Kopie). Das Hilfsmittel ist abzugeben. Sollte kein Hilfsmittel verwendet werden ist dies in der Klausur anzuzeigen.		**Dauer der Klausur:**	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. Achten Sie auf Schlüsselwörter wie „in Stichworten“, „kurz“, „ausführlich“ „nennen“, „erklären“ oder „im Detail“. Der Umfang Ihrer Antwort soll sich danach richten.

1. Allgemeines (8 Punkte)

a) Nennen Sie die allgemeine Definition eines Betriebssystems der Vorlesung?

2

b) Nennen Sie 3 unterschiedliche (Klassen von) Systemen auf denen Linux läuft

1.	
2.	
3.	

2

Name

c) Nennen Sie 2 unterschiedliche (Klassen von) Systemen auf denen Windows läuft?

1.	
2.	

1

d) Warum ist der Linux Kernel (hauptsächlich) in der Programmiersprache C programmiert?

2

e) Sie können die Datei `check` als User `me` nicht editieren:
`d-wx------  3 me  you 42 Feb 29 11:55 check`
Warum nicht – und wie können Sie den Fehler beheben?

hier ist kein r → keine Lesezugriffsberechtigung

1

2. Bash (16 Punkte)

a) Welche Vorteile haben Script-Sprachen wie Bash, welche Nachteile haben sie?

Vorteile? Source Code Plattformunabhängig, administrativ Lösen, leicht veränderbar
Nachteile? Geschwindigkeit, Bibliotheken, Typsicherheit

2

_____
Name

b) Was machen die folgenden Bash-Befehle. Bitte stichpunktartig erklären:

`du -h`	
`fsck`	
`dmesg`	
`objdump`	
`ps`	
`echo`	
`kill`	
`less`	

3

c) Geben Sie die passenden Shell Befehle an:

Tastendruck um Programm abzubrechen:	
Den Prozess 666 freundlicherweise auf Prio 10 setzen:	
Freien Plattenplatz menschenlesbar anzeigen lassen:	
Datei `f` in Verzeichnis `d` verschieben:	
Datei `b` anfassen (bspw. damit es ins Backup kommt!):	
Filesystem der 1. Partition auf der 2. Platte prüfen:	
Dateien in allen Verzeichnissen finden mit Endung `.h`:	
Symbolischer Link: `b` (Neu) zeigt auf `a` (Original):	

4

Name

d) Schreiben Sie das Bash Programm `count`: Es basiert auf dem Output von `last`:

```
rakeller pts/6 134.108.34.30     Wed Feb 29 11:55 still logged in
rakeller pts/7 134.108.34.67     Wed Feb 30 12:01 - 12:03 (00:03)
```

Schreiben Sie ein vollständiges Skript, welches die IP-Adresse als erstes Parameter nimmt und die Ausgabe von `last` nach dieser IP durchsucht und am Ende die IP und die Anzahl der Logins ausgibt. Eine Fehlerprüfung auf Anzahl Parameter ist Teil der Aufgabe – nicht aber Prüfung auf korrekte IPv4/IPv6 Adresse (dies ergibt aber Zusatzpunkte!)!

Die Ausgabe im obigen Fall wäre also:

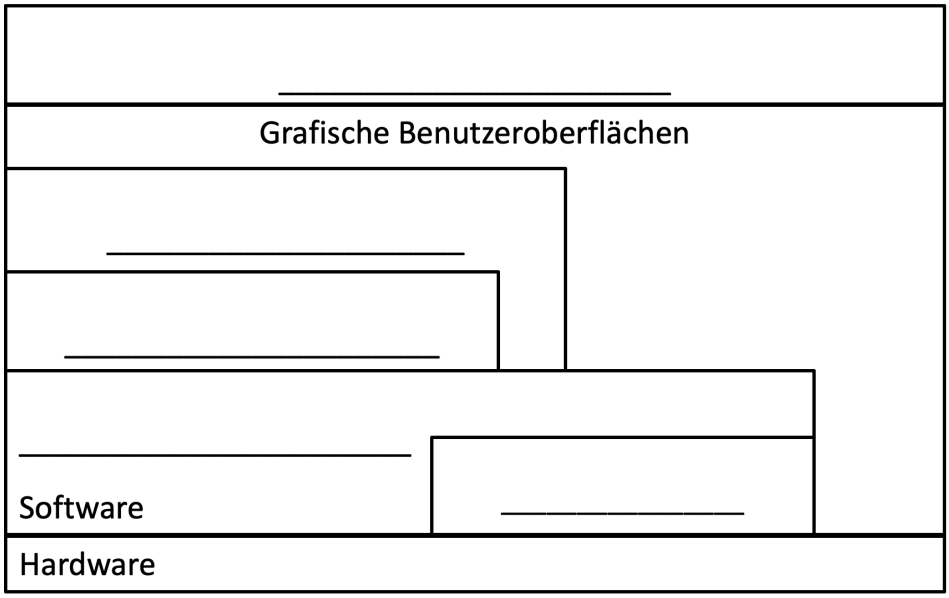
```
$ ./count 134.108.43.30↵

134.108.43.30 1
```

7

_____
Name

3. System Calls (7 Punkte)

a) Beschriften Sie die (SW-)Hierarchien auf modernen UNIX/Windows-Systemen:



2

b) Auf welche Arten kann der User-Kontext auf Intel x86 System-Calls aufrufen?

2

c) Ein HW-Interrupt tritt auf. Was passiert? Bitte den Ablauf kurz beschreiben!

3

Name

4. Linux Kernel & Scheduling (20 Punkte)

a) Was muss die HW mind. bieten, damit Scheduling implementiert werden kann?

1.	
2.	

2

b) In Linux Kernel arbeitet der Completely Fair Scheduler (CFS). Nach welchem Prinzip und mit welcher Datenstruktur arbeitet dieser?

2

c) Ein Prozess ruft einen blockierenden Systemcall (wie z.B. `read()` zum Lesen von Daten von der Festplatte). Was passiert? Erklären Sie die Schritte.

VFS = Kernel Subsystem

1.) Systemcall read(32

2.) Kernel Subsystem VFS prüft Daten vorhanden

3.) Entsprechende FS wird gerufen die Daten zu lesen

(4.) Gerätetreiber der Festplatte

ab hier dauert es (Interrupt) => wird schlafen gelegt

2

d) Welche (temporären) Informationen sollte ein Scheduler erfassen, um die Priorität eines Tasks neu zu berechnen?

2

_____
Name

e) Beurteilen Sie jede Aussagen ob diese Wahr (W) oder Falsch (F) ist:

Aussage	**W/F?**
Das 1:1 Modell reduziert Komplexität, ist damit einfacher	
Der Kernel weiß beim 1:1 Modell, daß ein Thread auf einen Lock wartet	
1:1 Modell heißt ein User-Thread entspricht einem Task im OS	
In `struct task{}` stehen die lauffähigen und gestoppten Tasks	
Der Scheduler wählt den Task mit der längstem `vruntime`	
Die `vruntime` ist in Millisekunden (ms) aufgeteilt	
Ein Task läuft nachdem er auf einem core gestartet wurde nur auf diesem	
Als Nutzer kann ich nicht die Priorität meiner Prozesse setzen	
Die CPU-Zeit wird an die `nr_running` Tasks verteilt	
Java Anwendungen mit mehreren Threads nutzen auf Linux nur 1 Core	

4

f) Geben Sie mind. vier Kommandos an, um den Linux Kernel zu konfigurieren, zu kompilieren und zu installieren.

1.	
2.	
3.	
4.	

2

Name

g) Programmieren Sie ein minimales Kernel-Module, welches beim Laden prüft, ob der ladende Task die FPU nutzt und dies auf der Debug-Konsole ausdruckt. Die relevante Datenstruktur `task_struct`, sowie die `PF_*` Flags (beide aus `include/linux/sched.h`) sind:

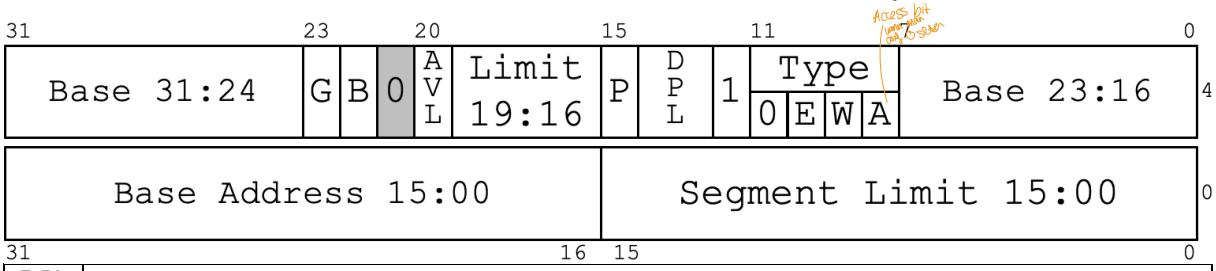
```
struct task_struct { ...
    volatile long state;    // -1 unrunnable, 0 runnable, >0 stopped
    unsigned int flags;     // Defined by PF_* below
... }
#define PF_IDLE       0x00000002
#define PF_USED_MATH  0x00002000    // If set, the FPU was used
```

6

Name

5. Virtueller Speicher (18 Punkte)

a) Wofür stehen und was bedeuten DPL, W, A und P im Deskriptor? (Zusatz: B)



DPL	
W	
A	
P	
B	2 Zusatzpunkte Big Boss

2

b) Was macht der TLB, wann wird er ausgelesen, wer darf auf ihn schreibend zugreifen?

Was macht der TLB?	Translation Lookaside Buffer
Wann gelesen?	bei jedem Speicherzugriff bevor ist er erlaubt?
Wer schreiben?	Kernel/BS

2

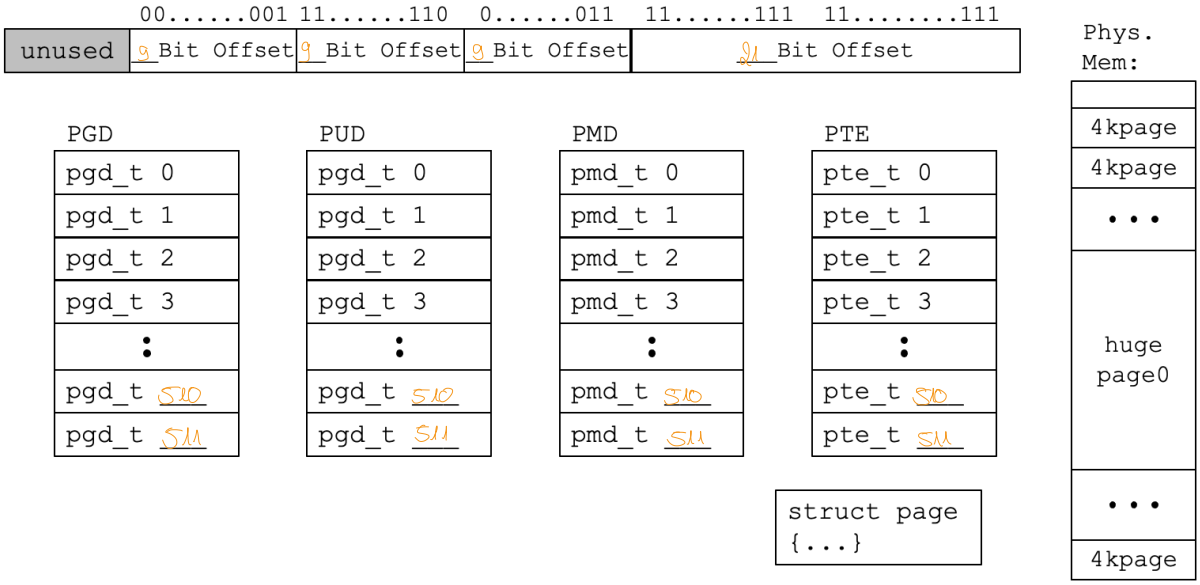
c) Was passiert bei einem TLB-Miss?

2

Name

d) Zeichnen Sie alle Pointer für die folgende 64-Bit Virtuelle Adresse ein – bitte **beachten** Sie die binären Zahlenwerte (0...001 bedeutet eine 1 im niederwertigsten Bit, ansonsten Nullen). Geben Sie weiterhin die Anzahl an den **unterstrichenen** Stellen ein.

Sollten Sie etwas korrigieren wollen, nutzen Sie bitte das Feld unten.



6

Platz für Kommentare und Verbesserungen

e) Für besonders große Server und Speicheranforderungen, wie wird diese Speicherhierarchie erweitert werden?

2

_____

Name

f) Der Buddy Allokator erlaubt, sehr effizient freie Speicherbereiche zu identifizieren. Die untenstehende Ansicht entspricht der Darstellung von Wikipedia. Im initialen Zustand 1 ist der gesamte Speicher frei. Zeichnen Sie die folgenden Allokationen ein:

1. Programm A allokiert 13 kB Speicher
2. Programm B allokiert 5 kB Speicher
3. Programm A gibt den Speicher wieder frei
4. Programm C allokiert 15 kB Speicher

	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB	4kB
1.	2^4															
2.																
3.																
4.																
5.																
6.																
7.																
8.																
9.																
10.																
11.																

4

Name

6. IPC & Synchronisation (9 Punkte)

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

Asynchrone Benach-richtigung eines Events:	
Gemeinsamer Speicher	
Uni-direktionaler Daten-transfer via Kernel	
Direkt in den Speicher eines Prozesses schreiben	

2

b) Nennen Sie Vor- und Nachteile Daten zwischen Prozessen mittels Dateien auszutauschen:

Vorteile: • Datei auf allen BS debuggen super • Signale • Message Queues • Sockets • Pipe
Nachteile: • debugger aber langsam

5

c) Nennen Sie vier Synchronisationsmechanismen:

1.	Monitore
2.	Mutexe
3.	Semaphore
4.	Locks

2

Name

7. Virtualisierung & Echtzeit-OS (12 Punkte)

a) Warum ist die x86-Architektur so besonders schwierig, effizient zu virtualisieren? (erinnern Sie sich an das Paper von VMware und die Virtualisierung mittels trap-and-emulate und eine der Instruktionen).

Instruktion wie popf verhalten sich anders je nach Ring 0 oder Ring 3 es ausgeführt wird.

3

b) Nennen Sie drei Vorteile von Virtuellen Maschinen – und drei Nachteile:

Vorteile:	
1.	Tools um viele Server/VMs schnell warten
2.	bessere Ausfallsicherheit
3.	einfache Reinstallation von Snapshot
Nachteile:	
1.	VM muss gepflegt werden
2.	erhöhte Kosten durch VM SW Lizenz
3.	VT kostet Performance

3

c) Nennen Sie drei Vorteile von Container-Technologien (z.B. Docker):

1.	
2.	
3.	
4.	

2

Name

d) Bei einem System gehen m periodische Ereignisse i mit der jeweiligen Periode P_i ein. Das Ereignis i verbraucht C_i CPU-Zeit. Wann ist dieses System noch stabil und kann diese Ereignisse zeitlich verarbeiten?

Geben Sie die Formel an.

2

e) Was heißt ein Standard Linux-Kernel (wie in der VL demonstriert) ist nicht echtzeitfähig. Was zeigte der demonstrierte Kernel?

Was heißt nicht echtzeitfähig:
Was zeigte dieser Kernel:

2