

**Aufgabe 1: (9 Punkte)**

Bei den Multiple-Choice-Fragen ist jeweils nur **eine** richtige Antwort eindeutig anzukreuzen. Auf die richtige Antwort gibt es die angegebene Punktzahl.

Wollen Sie eine Multiple-Choice-Antwort korrigieren, kreisen sie bitte die falsche Antwort ein und kreuzen die richtige an.

Lesen Sie die Frage genau, bevor Sie antworten.

a) Welche Aussage zum Thema symbolische Links und Hard-Links ist **falsch**? 2 Punkte

- Hard-Links sind nur innerhalb desselben Namensraumes möglich.
- Jeder symbolische Link verfügt über einen eigenen Inode, während Hard-Links alle auf denselben Inode verweisen.
- Symbolische Links können auf nicht mehr existierende Dateien derselben Partition verweisen.
- Jeder Verzeichniseintrag ist mit mindestens zwei symbolischen Links im Dateisystem eingetragen.

b) Man unterscheidet Traps und Interrupts. Welche Aussage ist **richtig**? 2 Punkte

**Hinweis: Sowohl Antwort (1) als auch Antwort (3) wurden als korrekt gewertet.**

- Bei der mehrfachen Ausführung eines unveränderten Programms mit gleicher Eingabe treten Traps immer an den gleichen Stellen auf.
- Das Auftreten eines Seitenfehlers beim Zugriff auf eine bestimmte Adresse ist zufällig und nicht vorhersagbar. Der Seitenfehler löst daher einen Interrupt aus.
- Der Zugriff auf eine physikalische Adresse kann zu einem Trap führen.
- Traps dürfen nicht nach dem Beendigungsmodell behandelt werden, da sich der Prozess beim Auftreten des Traps u.U. gerade in einem Systemaufruf aufhält und Datenstrukturen des Betriebssystems in einem inkonsistenten Zustand hinterlassen würde.

c) Welche Aussage zu den verschiedenen Gewichtsklassen von Prozessen **trifft zu**? 3 Punkte

- Federgewichtige Prozesse laufen im Gegensatz zu den leichtgewichtigen Prozessen im gleichen Adressraum und sind daher ressourcenschonender.
- Blockiert ein Benutzerfaden eines leichtgewichtigen Prozesses in einem Systemaufruf wie **read(2)**, so werden auch alle anderen Benutzerfäden des leichtgewichtigen Prozesses blockiert und müssen auf die Rückkehr des Systemaufrufs warten.
- Die Einplanung und Einlastung von Benutzerfäden federgewichtiger Prozesse findet immer ohne Wissen des Betriebssystems statt und erfordert daher entsprechende Infrastruktursoftware innerhalb der Anwendung selbst.
- Schwergewichtige Prozesse sind die einzige Klasse von Prozessen, die auf einem Multiprozessorsystem echt parallel ausgeführt werden kann, da nur hier jeder Benutzerfaden einem eigenen Kernfaden zugeordnet ist.

d) Welche Aussage im Bezug auf Adressräume **trifft zu**? 2 Punkte

- Die Adressumsetzung von logischen auf physikalische Adressen erfolgt immer zur Laufzeit durch eine spezielle Hardwareeinheit (MMU).
- In einem physikalischen Adressraum sind immer alle Adressen gültig.
- Ein virtueller Adressraum kann nur in Verbindung mit einem logischen Adressraum verwendet werden.
- Für physikalische Adressräumen existieren keine Schutzverfahren. Diese werden erst durch Verfahren wie Segmentierung in Kombination mit einem logischen Adressraum möglich.

**Aufgabe 2: (15 Punkte)**

*Sie dürfen diese Seite und die Manual-Seite am Ende der Klausur zur besseren Übersicht bei der Programmierung heraustrennen!*

Programmieren Sie eine Funktion **finddir**, die ein Verzeichnis (Pfadname in Parameter **pathname**) rekursiv durchläuft und die Pfade (**pathname** konkateniert mit dem dazu relativen Pfad) aller gefundenen Verzeichniseinträge ausgibt. Einträge, deren Name mit einem **.'** beginnt, werden hierbei komplett ignoriert, außer ein solcher Pfad wurde explizit beim **finddir**-Aufruf übergeben.

Schnittstelle:

```
int finddir(const char *pathname);
```

Sie können davon ausgehen, dass ein Pfadname niemals länger als **PATH\_MAX** Zeichen ist.

Rückgabewert:

0: die Ausgabe war erfolgreich

-1: bei der Operation ist ein Fehler aufgetreten

Anmerkung: Pfad-Konkationen können alternativ mit **strcpy(3)** und **strcat(3)** oder mit **sprintf(3)** durchgeführt werden!

```
/* includes */
#include <stdio.h>
#include <errno.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <dirent.h>
#include <limits.h>
```

**/\* Funktion finddir \*/**

	<input type="checkbox"/>

**Aufgabe 3: (6 Punkte)**

Beschreiben Sie jeweils kurz die Semantiken der drei Sendep primitiven  
*no-wait send*, *synchronization send* und *remote-invocation send*!

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....