

Aufgabe 1.1: Einfachauswahl-Fragen (22 Punkte)

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

Wollen Sie eine Multiple-Choice-Antwort korrigieren, streichen Sie bitte die falsche Antwort mit drei waagrechten Strichen durch (~~☒~~) und kreuzen die richtige an.

Lesen Sie die Frage genau, bevor Sie antworten!

- a) Man unterscheidet die Begriffe Programm und Prozess. Welche der folgenden Aussagen zu diesem Thema ist richtig? 2 Punkte
- Ein Prozess kann durch mehrere Programme ausgeführt werden.
 - Mit Hilfe des Systemaufrufs *exec* wird das bestehende Programm im aktuell laufenden Prozess ersetzt.
 - Der Binder erzeugt aus einer oder mehreren Objekt-Dateien einen Prozess.
 - Der Prozess ist der statische Teil (Rechte, Speicher, etc.), das Programm der aktive Teil (Programmzähler, Register, Stack).
- b) Man unterscheidet Traps und Interrupts. Welche Aussage ist richtig? 2 Punkte
- Das Betriebssystem kann Interrupts, die in ursächlichem Zusammenhang mit dem gerade laufenden Prozess stehen, nach dem Beendigungsmodell behandeln, wenn eine sinnvolle Fortführung des Prozesses nicht mehr möglich ist
 - Der Zugriff auf eine physikalische Adresse kann zu einem Trap führen.
 - Traps dürfen nicht nach dem Wiederaufnahmestapelmodell behandelt werden, da ein Trap immer einen schwerwiegenden Fehler signalisiert.
 - Ganzzahl-Rechenoperationen können nicht zu einem Trap führen.
- c) Welche Aussage zum Thema Threads ist richtig? 2 Punkte
- Zu jedem Kernel-Thread gehört ein eigener isolierter Adressraum.
 - Kernel-Threads können Multiprozessoren nicht ausnutzen.
 - Userlevel-Threads blockieren sich bei blockierenden Systemaufrufen gegenseitig.
 - Zur Umschaltung von Userlevel-Threads ist ein Adressraumwechsel erforderlich.

- d) Welche Aussage zum Thema Adressraumschutz ist richtig? 2 Punkte
- Beim Adressraumschutz durch Abteilung wird der logische Adressraum in mehrere Segmente mit unterschiedlicher Semantik unterteilt.
 - Adressraumschutz durch Eingrenzung benötigt keinen verschiebenden Lader, um Programmadressen an Arbeitsspeicheradressen zu binden
 - Beim Einsatz von Segmentierung ist es möglich, dass dieselbe logische Adresse in unterschiedlichen logischen Adressräumen auf unterschiedliche physikalische Adressen verweist.
 - In einem segmentierten Adressraum kann zur Laufzeit kein weiterer Speicher mehr dynamisch nachgefordert werden.
- e) Welche Aussage zum Thema Betriebsarten ist richtig? 2 Punkte
- Mehrzugangsbetrieb ist nur in Verbindung mit CPU- und Speicherschutz sinnvoll realisierbar.
 - Beim Stapelbetrieb können keine globalen Variablen existieren, weil alle Daten im Stapel-Segment (Stack) abgelegt sind.
 - Mehrprogrammbetrieb ermöglicht die simultane Ausführung mehrerer Programme innerhalb desselben Prozesses.
 - Echtzeitsysteme findet man hauptsächlich auf großen Serversystemen, die eine enorme Menge an Anfragen zu bearbeiten haben.
- f) Welche Aussage zum Thema Speicherverwaltung ist richtig? 2 Punkte
- Sollte bei einer Speicheranforderung mittels *malloc* nicht genügend Speicher vorhanden sein, wird der Prozess vom Betriebssystem beendet.
 - Ohne spezielle Unterstützung durch das Betriebssystem kann das Laufzeitsystem nur eine vorab statisch festgelegte Menge an Speicher feingranular verwalten.
 - Speicherbereiche, die vor Beendigung des Prozesses nicht mit *free* freigegeben wurden, sind bis zum Neustart des Systems unwiederbringlich verloren.
 - Speicherbereiche, die im logischen Adressraum zusammenhängend sind, müssen auch im physikalischen Hauptspeicher zusammenhängend sein.

- g) Welche Aussage zu Zeigern ist richtig? 2 Punkte
- Die Übergabesemantik für Zeiger als Funktionsparameter ist *call-by-reference*.
 - Ein Zeiger kann zur Manipulation von schreibgeschützten Datenbereichen verwendet werden.
 - Zeiger vom Typ `void*` existieren in C nicht, da solche "Zeiger auf Nichts" keinen sinnvollen Einsatzzweck hätten.
 - Zeiger können verwendet werden, um in C eine *call-by-reference* Übergabesemantik nachzubilden.
- h) Welche Aussage zum Thema RAID ist richtig? 2 Punkte
- RAID 5 ist nur mit drei Platten verwendbar, da die Berechnung der Paritätsinformationen bei mehr Platten nicht möglich ist.
 - Bei RAID 4 werden alle im Verbund beteiligten Platten gleichmäßig beansprucht.
 - Der Lesedurchsatz eines RAID-Systems mit mehreren Platten ist prinzipbedingt geringer als der Lesedurchsatz einer einzelnen Platte.
 - Bei RAID 1 werden die Datenblöcke über mehrere Festplatten verteilt und repliziert gespeichert.
- i) Welche Aussage zum Thema Schedulingverfahren ist richtig? 2 Punkte
- Round-Robin* bevorzugt E/A-intensive Prozesse zu Gunsten von rechenintensiven Prozessen.
 - Prioritäten-basierte Verfahren sind auch für interaktiven Betrieb geeignet.
 - First-Come-First-Served* ist nur im Stapelbetrieb bei lang laufenden Aufträgen sinnvoll einsetzbar.
 - Virtual-Round-Robin* benachteiligt E/A-intensive Prozesse zu Gunsten von rechenintensiven Prozessen.

- j) Für lokale Variablen, Aufrufparameter, etc. einer Funktion wird bei vielen Prozessoren ein Stack-Frame angelegt. Welche Aussage ist richtig? 2 Punkte
- Es ist nicht möglich auf lokale *automatic*-Variablen zuzugreifen, die sich außerhalb des eigenen Stack-Frames befinden.
 - Bei rekursiven Funktionsaufrufen kann der Speicher des Stack-Frames in jedem Fall wiederverwendet werden, weil die gleiche Funktion aufgerufen wird.
 - Ein Pufferüberlauf eines lokalen Feldes kann sicherheitskritisch sein, da er unter Umständen zur Ausführung beliebigen Codes führt.
 - Der Compiler legt zur Übersetzungszeit fest, an welcher Position im Stack-Frame der *main*-Funktion die globalen Variablen angelegt werden.
- k) Virtualisierung kann als Maßnahme gegen Verklemmungen genutzt werden. Warum? 2 Punkte
- Im Fall einer Verklemmung können zusätzliche virtuelle Betriebsmittel neu erzeugt werden. Diese können dann eingesetzt werden, um die fehlenden physikalischen Betriebsmittel zu ersetzen.
 - Durch Virtualisierung ist ein Entzug von physikalischen Betriebsmitteln möglich, obwohl dies auf der logischen Ebene unmöglich ist.
 - Eine Verklemmungsauflösung ist einfacher, weil virtuelle Betriebsmittel jederzeit ohne Schaden entzogen werden können.
 - Durch Virtualisierung kann man über Abbildungsvorgänge Zyklen, die auf der logischen Ebene vorhanden sind, auf der physikalischen Ebene auflösen.

Aufgabe 1.2: Mehrfachauswahl-Fragen (8 Punkte)

Bei den Multiple-Choice-Fragen in dieser Aufgabe sind jeweils m Aussagen angegeben, n ($0 \leq n \leq m$) Aussagen davon sind richtig. Kreuzen Sie **alle richtigen** Aussagen an. Jede korrekte Antwort in einer Teilaufgabe gibt einen halben Punkt, jede falsche Antwort einen halben Minuspunkt. Eine Teilaufgabe wird minimal mit 0 Punkten gewertet, d. h. falsche Antworten wirken sich nicht auf andere Teilaufgaben aus.

Wollen Sie eine falsch angekreuzte Antwort korrigieren, streichen Sie bitte das Kreuz mit drei waagrechten Strichen durch (~~☒~~).

Lesen Sie die Frage genau, bevor Sie antworten!

- a) Welche der folgenden Aussagen zum Thema Prozesszustände sind richtig? 4 Punkte
- Greift ein laufender Prozess lesend auf eine Datei zu und der entsprechende Datenblock ist nicht im Hauptspeicher vorhanden, dann wird der Prozess in den Zustand *bereit* überführt.
 - Bei kooperativem Scheduling ist kein direkter Übergang vom Zustand *laufend* in den Zustand *bereit* möglich.
 - Es können sich maximal genauso viele Prozesse gleichzeitig im Zustand *laufend* befinden, wie Prozessorkerne vorhanden sind.
 - Die V-Operation eines Semaphors kann bewirken, dass ein Prozess vom Zustand *blockiert* in den Zustand *bereit* überführt wird.
 - Im Rahmen der mittelfristigen Einplanung kann ein Prozess von Zustand *laufend* in den Zustand *schwebend laufend* wechseln.
 - Ein Prozess kann nur durch seine eigene Aktivität vom Zustand *laufend* in den Zustand *blockiert* überführt werden.
 - Bei Eintreffen eines Interrupts wird der aktuell laufende Prozess für die Dauer der Interrupt-Abarbeitung in den Zustand *blockiert* überführt.
 - Das Auftreten eines Seitenfehlers kann dazu führen, dass der aktuell laufende Prozess in den Zustand *beendet* überführt wird.

- b) Welche der folgenden Aussagen zu UNIX-Dateisystemen sind richtig? 4 Punkte
- In einem Verzeichnis darf es mehrere Einträge mit identischem Namen geben, sofern sie auf unterschiedliche *Inodes* verweisen.
 - Für das Löschen einer Datei sind die Rechte-Informationen im *Inode* des Verzeichnisses, das die Datei enthält, irrelevant.
 - Nach dem Löschen eines *Inodes* und der dazugehörigen Datenblöcke ist garantiert, dass kein *hard link* mehr darauf verweist.
 - Ein *Inode* kann im Dateisystem nicht über mehrere Namen referenziert werden.
 - Beim lesenden Zugriff auf eine Datei über einen *symbolic link* kann ein Prozess den Fehler *Permission denied* erhalten, obwohl er das Leserecht auf dem *symbolic link* besitzt.
 - Der *Inode* einer Datei wird getrennt von ihrem Inhalt auf der Platte gespeichert.
 - Ein *Inode* enthält u.a. die Anzahl der *symbolic links*, die auf ihn verweisen.
 - Die Anzahl der *hard links*, die auf ein Verzeichnis verweisen, hängt von der Anzahl seiner Unterverzeichnisse ab.

// Funktion showFrame
// Ausgabegeräte suchen

// Videobilder aus Puffer entnehmen und ausgeben

// Ende Funktion showFrame

```

/*****
 * Datei jitbuf.c
 *****/
#define CACHE_SIZE 100
#define BUFFER_SIZE 20

static volatile size_t readPos;
static volatile size_t writePos;
static SEM* fullSlots;
static SEM* freeSlots;
static SEM* waiter;
static size_t itemsBuffered;
static struct data cache[CACHE_SIZE];

// Funktion jbInit

```

```

// Funktion jbPut

```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

```

// Funktion jbGet

```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

```

// Funktion jbFlush

```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

J:

