

Probe. Klausur

12. Juli 2023

Vorname:	
Nachname:	
Matr.-Nr.:	
Studienfach:	

- Das Nutzen von **mitgebrachtem Papier** für die Lösungen ist **nicht gestattet!**
- Geräte für die Kommunikation und das Speichern und Anzeigen von Dokumenten sind **nicht zulässig**. Insbesondere zählen dazu Handys und Wearables. Schalten Sie diese aus und verstauen Sie diese in Ihrer Tasche. Auch sind keine Hilfsmittel, wie z.B. Taschenrechner, zugelassen!
- Nicht benötigte Sachen, wie Taschen und Jacken, legen Sie bitte **von Ihrem Platz entfernt** ab, z.B. im Bereich vor der Tafel. Verpflegung, Medikamente oder Ähnliches können Sie am Platz belassen.
- Nutzen Sie **dokumentenechte Stifte** mit **schwarzer** oder **blauer** Tinte, wie zum Beispiel Kugelschreiber. Mit Bleistift geschriebene Klausuren werden nicht akzeptiert.
- Halten Sie Ihren **Studenten-** und **Lichtbildausweis** bereit.
- Sie haben **120 Minuten** zum Lösen der Klausur.

- 1) Überprüfen Sie die Klausur auf **Vollständigkeit**. Sie hat 17 Seiten und ist doppelseitig gedruckt.
- 2) Schreiben Sie Ihren **Namen** und Ihre **Matrikelnummer** auf **jedes Blatt**.
- 3) Lesen Sie die Aufgabenstellung **aufmerksam** durch, bevor Sie mit dem Lösen beginnen. Melden Sie sich bei Unklarheiten.
- 4) Schreiben Sie Ihre Lösungen auf die Seiten der Klausur.
- 5) Benötigen Sie mehr Platz für Ihre Antworten, können Sie von uns extra Blätter bekommen. Machen Sie auf der Seite der Aufgabenstellung kenntlich, wo die Aufgabe (weiter) gelöst wird.
- 6) Verhalten Sie sich ruhig, auch, wenn Sie die Klausur fertig gelöst haben.
- 7) Geben Sie **jede** ausgehändigte Seite am Ende der Klausur ab!

Aufgabe	1	2	3	4	5	6	7	8	9	Σ
	<input type="checkbox"/>									
Punkte	14	12	17	15	14	10	14	14	10	120

Aufgabe 2: Prozesse und Threads

/12

a) Task A und Task B initialisieren ihren Speicher mit Nullen. Anschließend schreibt A den Wert 42 als `uint32_t` an die Adresse 1024. Darauf liest B den Wert im Speicher an der Adresse 1024 als `uint32_t`. Welchen Wert hat B gelesen?

/4

Falls A und B Prozesse sind:

Falls A und B Threads des gleichen Prozesses sind:

b) Beschreiben Sie den Ablauf eines Kontextwechsels von Prozess A zu Prozess B . Gehen Sie dabei insbesondere auch darauf ein, welche Informationen gespeichert werden.

/3

c) Beschreiben Sie den Ablauf eines Kontextwechsels von Thread A zu Thread B (des gleichen Prozesses). Gehen Sie dabei insbesondere auch darauf ein, welche Informationen gespeichert werden.

/3

d) Warum ist der Wechsel zwischen Threads (des gleichen Prozesses) im Vergleich zum Kontextwechsel zwischen Prozessen schneller?

/2

Aufgabe 3: Prozess-Scheduling

Sie haben verschiedene Verfahren der Zuteilung von CPU-Zeit zu einzelnen Prozessen in einem Multitasking-Betriebssystem kennengelernt.

Folgende fünf Prozesse seien gegeben:

Prozess	Ankunftszeit	Ausführungszeit
A	0	3
B	2	4
C	4	4
D	6	3
E	8	1

a) Wenden Sie den Scheduling Algorithmus *Round Robin* auf das obige Beispiel an. Markieren Sie in der Tabelle den zum jeweiligen Zeitpunkt laufenden Prozess mit einem Kreuz in der zugehörigen Spalte. Wenn zwei Prozesse in Frage kommen, wird der alphabetisch kleinere Prozess gewählt.

Zeit	Prozess					Running	Queue
	A	B	C	D	E		
0	x					A	
1	x					A	
2		x				B	A
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							

Name:

Matr.-Nr.:

b) Zählen Sie alle aus der Vorlesung bekannten Scheduling Algorithmen auf und beschreiben Sie jeden mit maximal einem Satz.

/5

c) Geben Sie einen Vorteil und einen Nachteil zu jedem Scheduling Algorithmus an.

/5

Name:

Matr.-Nr.:

d) Beschreiben Sie die das 1. Reader-Writer-Problem, das 2. Reader-Writer-Problem und das 3. Reader-Writer-Problem.

/1

e) Für welche der Reader-Writer-Problem kann es zu Starvation kommen? Wenn ja, unter welchen Umständen tritt Starvation auf?

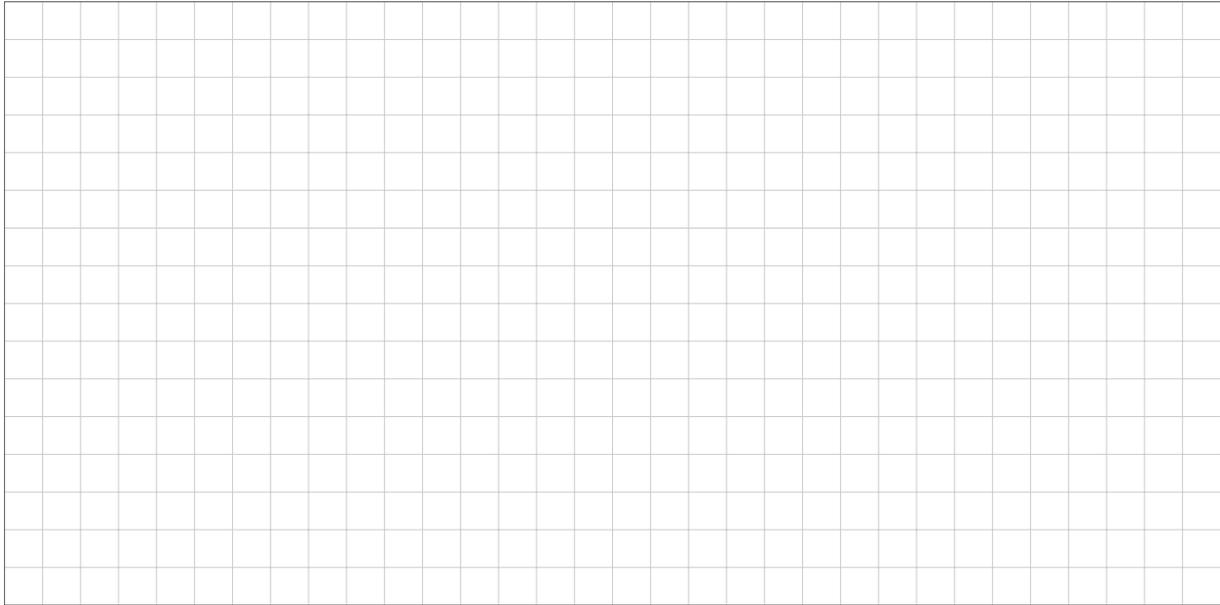
/2

Name:

Matr.-Nr.:

c) Lassen sich Zugriffsrechte (wie nur lesbar, ausführbar, nicht ausführbar) besser auf Segmentbasis oder auf Seitenbasis implementieren?

/2



d) Welche der beiden Varianten des unten stehenden Programms wird die höhere Cache-Effizienz haben? Begründen Sie Ihre Entscheidung.

Hinweis: Gehen Sie davon aus, dass `length` sehr groß ist und der Compiler keine Optimierungen am Programm vornimmt.

Listing 1: Variante A

```
1 void do_calculation(uint32_t *inout, size_t length)
2 {
3     for (size_t i = 0; i < length; i++) {
4         inout[i] ^= 0x55555555;
5         inout[i] = inout[i] * 1337 / 42;
6     }
7 }
```

Listing 2: Variante B

```
1 void do_calculation(uint32_t *inout, size_t length)
2 {
3     for (size_t i = 0; i < length; i++) {
4         inout[i] ^= 0x55555555;
5     }
6
7     for (size_t i = 0; i < length; i++) {
8         inout[i] = inout[i] * 1337 / 42;
9     }
10 }
```



/14

Aufgabe 8: Ein- und Ausgabesysteme

/3

a) Erläutern Sie die Begriffe „programmierte E/A“, „interrupt-getriebene E/A“ und „DMA“.

/3

b) Erläutern Sie den Zusammenhang von Interrupts und DMA.

/1

c) Erläutern Sie das Konzept von Double-Buffering anhand eines Anwendungsbeispiels.

Aufgabe 9: Verständnisfragen

/10

Kreuzen Sie für alle Aussagen an, ob diese richtig oder falsch ist.

Hinweis: Ist eine Aussage nicht in jeglicher Hinsicht richtig, dann ist sie falsch.

Hinweis zur Bewertung:

+1 Punkt bei korrekter Antwort

-1 Punkt bei falscher Antwort

0 Punkte bei keiner Antwort.

Negative Gesamtsummen ergeben für diese Aufgabe 0 Gesamtpunkte.

- | | true | false |
|------------------------------------------------------------------------------------------------------------------------------|-----------------------|-----------------------|
| a) Zeit, Speicher und Energie sind primäre Leistungsmetriken für ein Betriebssystem. | <input type="radio"/> | <input type="radio"/> |
| b) Beim arithmetischen Überlauf werden Interrupts weder direkt von der Hardware noch direkt von der Software ausgelöst. | <input type="radio"/> | <input type="radio"/> |
| c) Die effektive Ausnutzung der Betriebsmittel gehört zu den Aufgaben eines Betriebssystems. | <input type="radio"/> | <input type="radio"/> |
| d) Das Text-Segment eines Programms kann im Allgemeinen als nur lesbar markiert werden. | <input type="radio"/> | <input type="radio"/> |
| e) Der Thread-Wechsel erfordert bei User-Level Threads keinen Wechsel den Kernel-Mode. | <input type="radio"/> | <input type="radio"/> |
| f) Der kritische Bereich bezeichnet die Phase, in denen ein Prozess / Thread auf gemeinsam genutzten Speicher zugreift. | <input type="radio"/> | <input type="radio"/> |
| g) Ein Mutex ist als Synchronisierungsmechanismus funktional gleichwertig zur Semaphore. | <input type="radio"/> | <input type="radio"/> |
| h) Für ein Deadlock müssen erfüllt sein: „Mutual Exclusion“, „Hold and Wait“, „No Preemption“ und „Circular Wait“ | <input type="radio"/> | <input type="radio"/> |
| i) Bei der Segmentierung kann zwar externe Fragmentierung auftreten, jedoch keine interne Fragmentierung. | <input type="radio"/> | <input type="radio"/> |
| j) Durchsatz, Antwortzeit und Prozessor-Effizienz (optimale Auslastung der CPUs) sind primäre Ziele von Schedulingverfahren. | <input type="radio"/> | <input type="radio"/> |