

(Prüfungs-)Aufgaben zum Thema Prozesse und Threads

Mit einem **D** gekennzeichnete Aufgaben sind nur für die Prüfung im Diplomstudiengang relevant.

1) Geben Sie die drei prinzipiellen Zustände eines Prozesses und die möglichen Übergänge zwischen diesen Zuständen an (z.B. durch eine Skizze) und beschreiben Sie, wodurch die einzelnen Zustandsübergänge ausgelöst werden.

2)

a) Geben Sie die drei prinzipiellen Zustände eines Prozesses und die möglichen Übergänge zwischen diesen Zuständen an (z.B. durch eine Skizze).

b) Wodurch können die einzelnen Zustandsübergänge ausgelöst werden?

3D) Die meisten Betriebssysteme (z.B. auch VMS und UNIX) verwenden zwei separate Datenstrukturen zur Beschreibung eines Prozesses. Worin unterscheiden sich diese beiden Datenstrukturen?

Geben Sie für jede dieser beiden Datenstrukturen drei Prozeßparameter an, die in ihnen enthalten sind, und begründen Sie, warum der jeweilige Parameter in dieser Datenstruktur liegt.

4) Welche Gründe gibt es für die Implementierung von Threads?

5) Beschreiben Sie den Ablauf einer Prozeßerzeugung an einem UNIX-System.

6) Beschreiben Sie den prinzipiellen Ablauf der Ereignisse an einem Rechnersystem, das mit Prozessen und Interrupts arbeitet.

7) Erklären Sie das Konzept eines Threads (insbesondere in Abgrenzung zu dem eines Prozesses).

9) Was ist ein Thread? Erklären Sie insbesondere auch den Zusammenhang mit und die Unterschiede zu einem Prozeß.

10) Erläutern Sie die Funktion und den prinzipiellen Ablauf des System Calls `exec(...)` an einem Unix-System. Welche Angabe(n) muß (müssen) in dem (den) Parameter(n) auf jeden Fall enthalten sein?

11) Erläutern Sie die Verwendung von Interrupt-Prioritäten in einem Betriebssystem.

12) Erläutern Sie die Schritte, die beim Auftreten eines Interrupts zu dessen Behandlung ausgeführt werden. (Hinweis: Denken Sie dabei auch an die Priorisierung der Interrupts.)

13) Was sind die Vorteile, wenn ein Betriebssystem Prozesse in Threads unterteilt?

14) Wie verwendet ein Betriebssystem Interrupt-Prioritäten zur Synchronisation des Zugriffes auf Datenstrukturen?

15) Sie sollen ein Betriebssystem, das bisher nur mit Prozessen arbeitet so umschreiben, daß die Prozesse in Threads unterteilt werden. Offensichtlich müssen Sie die Datenstruktur, die einen Prozeß beschreibt, aufteilen in eine Datenstruktur, die weiterhin den Prozeß als Ganzes beschreibt und eine thread-spezifische Datenstruktur.

Geben Sie jeweils drei Datenfelder bzw. Datenstrukturen an, die in der prozeßweiten Datenstruktur verbleiben, bzw. die in die thread-spezifische Datenstruktur verlagert werden.

prozeßweit	thread-spezifisch

16) Beschreiben Sie die Schritte vom Auftreten eines Interrupts bis nach dessen Abarbeitung.

17) Erläutern Sie die Begriffe "Prozeß" und "Thread" (in einem Betriebssystem, das Prozesse in Threads unterteilt).

18)

a) Geben Sie die drei prinzipiellen Zustände eines Prozesses (bzw. Threads) und die möglichen Übergänge zwischen diesen Zuständen an (z.B. durch eine Skizze).

b) Wodurch können die einzelnen Zustandsübergänge ausgelöst werden?

c) Von den 6 möglichen Zustandsübergängen kommen nicht alle vor. Erklären Sie für jeden fehlenden Übergang, warum dieser nicht auftritt. (Einer davon könnte in einer speziellen Situation auftreten, die in der Vorlesung der Einfachheit halber nicht erwähnt wurde: 1 Zusatzpunkt)

19)

a) Was ist der Unterschied zwischen der Ausführung von Code im User-Mode und der Ausführung im Kernel-Mode?

b) Wie ist festgelegt, in welchem Modus gerade gearbeitet wird?

c) Welche Möglichkeit(en) gibt es, in den Kernel-Mode zu gelangen?

20D)

a) Erläutern Sie, was man unter User-Threads und was unter Kernel-Threads versteht.

(Mit „Kernel-Thread“ ist hier dasselbe gemeint wie mit „Thread“ in der Vorlesung oder mit „lightweight process“ in vielen Unix-Systemen, nicht die leichte Variante davon wie z.B. unter Solaris. Gehen Sie nicht auf die programmiertechnischen Schwierigkeiten bei der Benutzung von User Threads ein.)

b) Was sind die prinzipiellen Vor- und Nachteile von User Threads?

21) Beschreiben Sie die Erzeugung eines Prozesses unter Unix und unter einem anderen Betriebssystem (z.B. Windows). Vergleichen und bewerten Sie diese beiden unterschiedlichen Vorgehensweisen.

22)

a) Ein Betriebssystem unterteile Prozesse in Threads. Erläutern Sie die Begriffe "Prozess" und "Thread" und ihr Verhältnis zueinander.

b) Wie kann ein Programmierer weitere Threads in einer Applikation erzeugen? (Ein Stichwort genügt.)

c) Worauf muss beim Programmieren einer multithreaded Applikation besonders geachtet werden?

23)

Wie reagiert ein Betriebssystem auf das Auftreten eines Interrupts ?

a) Beschreiben Sie die einzelnen Schritte des Betriebssystems in der zeitlichen Reihenfolge ab Auftreten des Interrupts bis zu dessen vollständiger Abarbeitung. Dabei sei zunächst angenommen, dass zwischenzeitlich keine weiteren Interrupts auftreten.

b) Was geschieht, wenn während der Behandlung eines Interrupts ein weiterer Interrupt auftritt (ein bis zwei Sätze) ?

c) Was geschieht, wenn ein Thread niedriger Priorität durch einen Interrupt unterbrochen wird, und während der Behandlung des Interrupts ein Thread höherer Priorität bereit gemacht, also in den Ready-Zustand gesetzt wird?

24) Beschreiben Sie einen möglichen Weg eines Threads vom Zustand „bereit“ über die Zustände „laufend“ und „blockiert“ zurück in den Zustand „bereit“. Geben Sie jeweils an, wodurch und/oder von wem der Zustandsübergang ausgelöst wird.

25) Ordnen Sie die folgenden Interrupts aufsteigend nach ihren Interrupt-Prioritäten:

- I/O Completion Interrupt
- Clock-Interrupt
- Scheduler Interrupt
- Power Failure Interrupt

26)

d) Was versteht man unter „Skalierbarkeit“ bei SMP-Systemen?

e) Welcher Teil des Hardware-Designs eines SMP-Systems ist der größte Engpass und beeinträchtigt die Skalierbarkeit am meisten?

f) Welche Aufgabe beim Design des Betriebssystems hat die größte Auswirkung auf die Skalierbarkeit eines SMP-Systems? (2-3 Sätze, keine Details.)

27) Welche Messgröße (wie sie z.B. im System Monitor oder im Process Explorer unter Windows oder im proc-Dateisystem unter Linux angezeigt wird) ist geeignet, um auch geringfügige Aktivitäten eines Threads festzustellen? Welche Messgröße, die ebenfalls unmittelbar mit der Thread-Aktivität zu tun hat, ist dafür i.a. ungeeignet, und warum?

28) Wieviele Threads mit Standard-Parametern kann ein Prozess unter Windows ungefähr erzeugen? Warum ist dies so?

29)

a) Nennen Sie drei Aufgaben, die vom Clock Interrupt Handler ausgeführt werden.

b) Wann wird vom Clock Interrupt Handler der Scheduler ausgelöst?