

CAuBS Einsendeaufgaben 3 – Deadline 01.01.2018 23:55 Uhr



Aufgabe 1: 3.2.11.2.1 Aktives Warten: Der Nachteil des aktiven Wartens

Welchen großen Nachteil besitzen alle Verfahren, die nach dem Prinzip des aktiven Wartens verfahren?

Es wird Rechenzeit mit dem Warten verbrannt.



Aufgabe 2: 3.2.11.3 Semaphore: P() und V() sind atomar

Die P()- und V()-Operation ist jeweils als atomare Operation definiert. Was bedeutet dies für die Ausführung dieser beiden Operationen?

Die Operationen dürfen nicht durch einen Kontextwechsel unterbrochen werden und müssen am Stück ausgeführt werden um einen Kontextwechsel zwischen Prüfung und Setzen des Locks zu vermeiden.



Aufgabe 3: 3.2.12 Deadlocks: Deadlock-Philosophen

Mandl 2013 geht am Ende von Kapitel 6.2.2 auf das Philosophenproblem und eine dabei bestehende Deadlock-Gefahr ein.

Erläutere:

- Unter welcher Bedingung tritt bei den speisenden Philosophen ein Deadlock-Zustand ein?

Wenn alle Philosophen gleichzeitig die linke Gabel aufnehmen, warten sie daraufhin unendlich lange auf die rechte Gabel

- Welche Rolle spielt eine atomare Aktion dabei?

Es würde kein Deadlock auftreten wäre das Aufnehmen beider Gabeln ununterbrechbar wäre

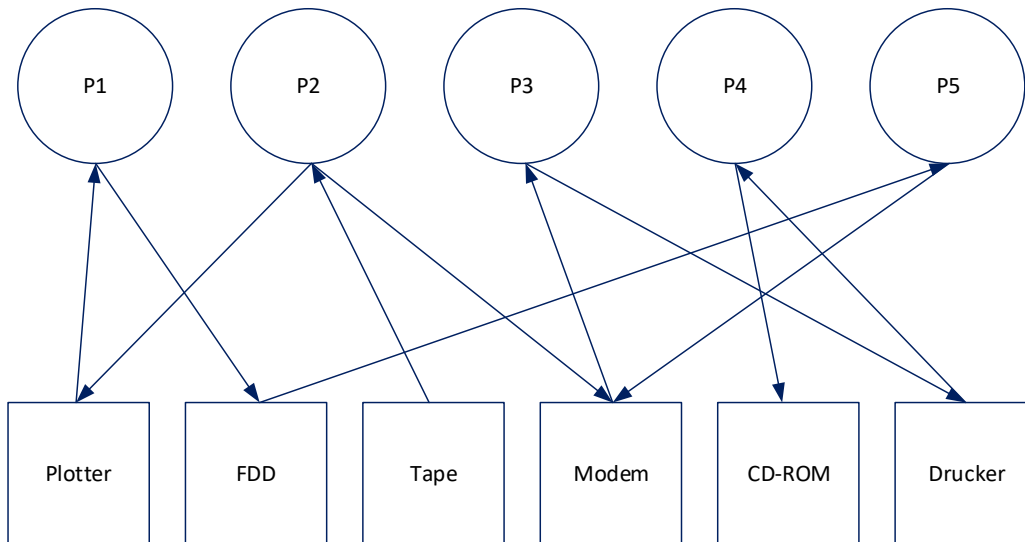


Aufgabe 4: 3.2.12.2 Deadlocks erkennen: Zeichne den Betriebsmittelgraph und erkenne den Deadlock!

Gegeben seien die Prozesse P1 bis P5 und die Ressourcen Drucker, Plotter, Modem, Magnetbandlaufwerk sowie Diskettenlaufwerk und CD-ROM-Laufwerk. Jede Ressource sei genau einmal vorhanden. Aktuell besteht folgende Ressourcenzuteilung und -anforderung:

- P1 belegt Plotter und fordert Diskettenlaufwerk an.
- P2 belegt Magnetbandlaufwerk und fordert Plotter und Modem an.
- P3 belegt Modem und fordert Drucker an.
- P4 belegt Drucker und fordert CD-ROM-Laufwerk an.
- P5 belegt Diskettenlaufwerk und fordert Magnetbandlaufwerk an.

- Zeichne den Betriebsmittelgraphen! Achte dabei auf die Pfeilrichtungen!



Offensichtlich liegt hier ein Deadlock vor.

- b. Woran ist dies im Betriebsmittelgraphen erkennbar? Welche Prozesse sind daran beteiligt?

Es ist dadurch zu erkennen, dass alle Betriebsmittel an Prozesse angebunden sind, die wiederum weitere Betriebsmittel anfordern um fortfahren zu können; die belegten Betriebsmittel werden somit niemals freigegeben.



Aufgabe 5: 3.2.12.4 Deadlocks vermeiden: Spooling

Recherchiere und erläutere: Was versteht man unter Spooling? (Zum Beispiel bei einem Drucker-Spooler.) Spooling ist prinzipiell geeignet, um Deadlocks zu vermeiden. Welche der vier Bedingungen macht Spooling unerfüllbar?

Der Druckauftrag wird nicht an den Drucker sondern dem Spooler übergeben und dort in die Warteschlange eingereiht. Somit ist der Auftrag für den Prozess erledigt und nur der Spooler muss exklusiv auf den Drucker zugreifen. Ein direkter Kontakt zum Drucker findet nicht statt. Daher sind die Bedingungen „Eine Ressource steht einem Prozess nur exklusiv zur Verfügung“, „Prozesse warten und behalten die Kontrolle“, „Ressourcen können nicht entrissen werden“ und auch „zyklische Kette“ für die gespoolte Komponente nicht erfüllt.



Aufgabe 6: 3.2.13 Interprozesskommunikation: IPC-Grundbegriffe

Mandl 2013 erläutert in Kapitel 6.5.1 einige Grundbegriffe der Kommunikation. Was versteht man demnach unter:

- a. Verbindungsorientierter/verbindungsloser Kommunikation

Der Unterschied ist am ehesten anhand von TCP/UDP nachzuvollziehen. Bei verbindungsloser Kommunikation (UDP) werden die Daten ans Ziel gesendet und ohne Rückmeldung und weitere Prüfung angenommen, dass diese schon größtenteils in Ordnung sein werden. Dies ist sehr gut für Echtzeitvideos geeignet, wo rechtzeitiges Eintreffen wichtiger ist als ein jederzeit 100% perfektes Bild. Bei verbindungsorientierter Kommunikation wird eine Verbindung aufgebaut, die Daten werden gesendet und deren Empfang bestätigt. Dies macht bei wichtigen Daten mehr Sinn als die schnellere verbindungslose Kommunikation – etwa die Brennstabtemperatur im Kernkraftwerk sollte nicht nur meistens richtig übermittelt werden oder nur vielleicht ankommen.

b. Speicherbasierter/nachrichtenbasierter Kommunikation

Bei speicherbasierter Kommunikation gibt es immer ein gemeinsames Speichermedium, in dem die Prozesse Daten austauschen können. Abgesehen von Shared Memory, was dafür aufgrund der wenigen beteiligten Komponenten einen enormen Geschwindigkeitsvorteil hat, ist die speicherbasierte Kommunikation meist sicherer und nachvollziehbarer/Rollbackfähig und wird daher gerne bei kritischen Anwendungen angewendet (Quorum im Failovercluster, Datenbanklogs)

Bei nachrichtenbasierter Kommunikation werden Mechanismen des Betriebssystems genutzt, mit denen sich die Prozesse direkt Nachrichten schicken können ohne dass ein zusätzlicher Speicher benötigt wird. Sofern es möglich ist, bietet es sich an, bei der Softwareplanung Sockets bzw. TCP/UDP zu verwenden falls die Anwendung später auf mehrere Server skaliert werden soll. Muss ein Prozess dem anderen Prozess ständig Daten übergeben, eignen sich eher lokale Mechanismen wie Pipes.

Message Queues gehören zur nachrichtenbasierten Kommunikation, sind aber im Grunde eine Mischung aus beidem Konzepten, da hier Nachrichten in die Queue gestellt und abgerufen werden können.

c. Synchroner/asynchroner Kommunikation

Bei synchroner Kommunikation wird immer blockiert bis der Partner den Empfang der Nachricht bestätigt hat bzw. der Empfänger wartet auf die Nachricht bevor er etwas anderes macht. Synchrone Kommunikation ist in der Regel sicherer (z.B. der Festplatten-Controller nimmt erst neue Daten an wenn die Festplatte das Schreiben bestätigt hat – „write trough“) während asynchrone Kommunikation meist deutlich schneller ist (der Festplattencontroller hat mehrere GB RAM und puffert und schreibt auf die Festplatte wenn er gerade Zeit dazu hat – „write back“)

d. Halbduplex-Betrieb/ Vollduplex-Betrieb

Im halbduplex kann jeweils nur gesendet oder empfangen werden während im Vollduplex beides möglich ist.

e. Unicast/Multicast/Anycast/Broadcast

Unicast ist die klassische Kommunikation mit nur einem Ziel

Multicast sendet an eine Gruppe von Empfängern und eignet sich gut um große Datenmengen wie z.B. Betriebssystemimages an mehrere Empfänger zu versenden. Der Sender verbraucht dabei nur einmal Bandbreite. Da alle Hardware auf dem Weg IGMP kompatibel sein muss, ist hier bei der Verwendung im kleineren Bereich immer Aufwand/Nutzen abzugleichen.

Anycast wird verwendet um mehrere Systeme unter der gleichen Adresse bereitstellen zu können. Etwa Googles DNS Server 8.8.8.8 wird nicht immer den gleichen Server erreichen sondern den, der dem Routing nach am günstigsten liegt.

Broadcast versendet an alle im Netzbereich/alle erreichbaren Prozesse



Aufgabe 7: 3.3.1 Virtuelle Speicherverwaltung: Rahmen vs. Seite

Was ist der Unterschied zwischen einem *Seitenrahmen* und einer *Seite*?

Anhand des [Videos](#) kann nachvollzogen werden, warum die Größe eines *Seitenrahmens* und einer *Seite* stets identisch sein muss. Erläutere warum!

Ein Seitenrahmen bezieht sich auf den Platz im physikalischen Speicher, in dem eine „virtuelle“ Speicherseite abgelegt wird. Da eine Seite genau in einen Seitenrahmen passen muss, sind diese gleich groß.



Aufgabe 8: 3.3.2 Swapping und Paging: Paging und die Seitentabelle

Eine (virtuelle) Seite befinde sich derzeit in einem Seitenrahmen eingelagert. Das Betriebssystem möchte diese Seite nun in den Hintergrundspeicher auslagern.

- Wie sieht der Seitentableneintrag für die betreffende Seite aus, solange die Seite noch eingelagert ist?

Das Present/Abent-Bit steht auf 1 sodass die MMU die Daten nicht wieder in den Speicher einlagern muss. (kein Seitenfehler). So muss nur der virtuelle Adressteil durch den physikalischen ersetzt werden.

- Welche Änderung muss das Betriebssystem an dem betreffenden Seitentableneintrag vornehmen, sobald die Seite ausgelagert wurde?

Das Present/Abent-Bit muss auf 0 gesetzt werden. Wird die Seite wieder benötigt, muss sie vom Betriebssystem wieder eingelagert werden.



Aufgabe 9: 3.3.2.2.3.2 NRU - Not Recently Used Algorithmus: Auch das M-Bit löschen?

Wenn nach Ablauf der erwähnten Zeitspanne das R-Bit gelöscht wird, sollte dann gleich auch das M-Bit mit gelöscht werden? Begründe deine Meinung!

Das M-Bit darf nicht gelöscht werden, da anhand dessen entschieden wird, ob die Seite einfach weggeworfen werden kann oder gesichert werden muss.



Aufgabe 10: 3.3.2.2.3.3 FIFO - First In First Out Algorithmus: FIFO und die Seitentabelle

Angenommen, der FIFO-Algorithmus wird in einem Betriebssystem eingesetzt. Wie sieht dann ein Seitentableneintrag aus? Werden insbesondere das R-Bit und das M-Bit benötigt?

Das R bit wird nicht benötigt, da stets die am längsten eingelagerte Seite entsorgt wird. Das M-Bit kann sinnvoll sein um festzustellen, ob die Seite weggeworfen werden kann oder erst gesichert werden muss. Es wird nicht benötigt wenn pauschal immer gesichert wird. Am Anfang der Seitentabelle steht die älteste Seite und am Ende werden neue Seiten angehängt.



Aufgabe 11: 3.3.2.2.3.5 Working Set Algorithmus: Die 80/20-Regel und das Working Set

Du kennst sicher die 80/20-Regel:

80 Prozent der von einer Software bereitgestellten Funktionen werden höchstens von 20 Prozent der Nutzer tatsächlich eingesetzt.

Erläutere (unter der Voraussetzung, dass die Regel zutrifft): Wie unterstützt diese Regel den Working Set Algorithmus bei seinem Ziel, möglichst wenig Seitenfehler entstehen zu lassen?

Wenn nur die zuletzt referenzierten Seiten im Speicher gehalten werden, kommt das der 80/20 Regel am nächsten. Etwa ist zu erwarten, dass Zeiterfassungsprogramme wie Calitime ständig Zeiten für Mitarbeiter buchen aber nur selten eine Monatsabrechnung durchführen. Aufgrund der ständigen Zugriffe auf „Zeit buchen“ ist somit diese im Speicher während die Monatsabrechnung recht schnell ausgelagert wird.

Ein Programm mit völlig zufälligen Datenzugriffen dürfte kaum profitieren.



Aufgabe 12: 3.4.1 Rolle der Geräteverwaltung: Systemaufruf zwischen User- und Kernel-Mode

Betrachte noch einmal das Kapitel Kernel-Mode, User-Mode und Systemaufrufe und wiederhole (in eigenen Worten), was es mit eben diesen Fachbegriffen auf sich hat:

- a. Kernel-Mode

Im Kernel-Mode darf die Hardware angesprochen werden und es stehen erweiterte Befehle zur Verfügung.

- b. User-Mode

Im Usermode werden Prozesse ohne direkten Zugriff auf Systembefehle ausgeführt.

- c. Systemaufruf

Bei einem Systemaufruf bittet der Prozess im Usermode das Betriebssystem darum, auf Gerät XY z.B. einen Ausdruck auszulösen da der Usermode Prozess dazu kein Recht hat.



Aufgabe 13: 3.4.4 Interruptbehandlung für ein Gerät: Geräte und ihre Interrupts

Betrachte die folgenden Geräte:

- Maus
- Tastatur
- Netzwerkkarte

und beantworte die folgenden Fragen:

- a. Ist das Gerät DMA-fähig?

Nur die Netzwerkkarte ist DMA fähig. Aufgrund der minimalen Datenmengen bei Eingabegeräten wäre DMA auch nicht sinnvoll.

- b. Wann tritt bei der Arbeit mit dem Gerät jeweils ein Interrupt auf? (Nenne nur die wichtigsten Gründe.)

Maus/Tastatur: Benutzereingabe

Bei der Netzwerkkarte

- *wenn sie genug Pakete gesammelt hat um Daten zu übertragen und die Systemlast das zulässt (Interrupt Moderation bei LAN-Karten)*
- *Die Netzwerkkarte sonstige Meldungen absetzt*

- c. Welche Schritte oder Tätigkeiten muss die Interruptbehandlungsroutine im zugehörigen Treiber ausführen?

Die Interruptbehandlungsroutine muss die Daten des Gerätes annehmen, weitergeben und dann die Routine beenden.



Aufgabe 14: 3.5.2 Dateisystem: CRUD für Verzeichnisse?

Lassen sich im Hinblick auf die *Verwaltung von Verzeichnissen* auch CRUD-Operationen angeben? Erläutere!

Bei Verzeichnissen müssten die gleichen Operationen wie bei Dateien möglich sein, da auch diese angelegt, aktualisiert gelesen und gelöscht werden können.



Aufgabe 15: 3.5.5.1 FAT - File Allocation Table: Speichere eine neue Datei!

Betrachte die oben gegebene Beispiel-FAT. Eine neue Datei *abc.mp3* soll gespeichert werden. Aufgrund der Dateigröße werden insgesamt fünf Cluster zum Speichern benötigt.

- a. In welchem Cluster beginnt die Datei? (Wähle eine geeignete Cluster-Nr.!)

Die Datei beginnt im ersten freien Cluster 4

- b. In welchen weiteren Clustern legst du die Datei ab?

In allen freien Clustern

Wie ändern sich also die Einträge in der FAT?

4 verweist auf 7, 7 auf 10, 10 auf 13 usw. und endet mit „nil“ wenn die Datei zu Ende ist.