



System Architektur (Architecture)

Klausur (Final Examination)

WS 1999/2000

14. April 2000

- Bitte tragen Sie zuerst auf allen Klausurblättern Ihren Namen, Vornamen und Ihre Matrikelnummer ein, auch auf den Konzeptblättern. *Please enter your last name, first name and matriculation number on each page (including used and unused draft pages).*
- Die Prüfung dauert 60 Minuten und besteht aus 5 Aufgaben auf 11 Seiten und zwei Konzeptblättern. *You have 60 minutes to complete your answers. The examination consists of 5 questions on 11 pages. You receive two additional blank pages for drafts, etc.*
- Die Prüfung ist mit mindestens 20 Punkten von 60 erreichbaren Punkten bestanden. *You pass the examination by obtaining at least 20 marks out of the possible 60 marks.*
- Es sind keinerlei Hilfen erlaubt! *No additional means are allowed!*
- Die Prüfung gilt als nicht bestanden, wenn Sie versuchen, aktiv oder passiv zu betrügen. *You fail the examination if you try to cheat actively or passively.*
- Wenn Sie zusätzliches Konzeptpapier benötigen, verständigen Sie bitte die Klausuraufsicht. *If you need more draft pages please notify one of the supervisors.*
- Bitte machen Sie eindeutig klar, was Ihre endgültige Lösung zu den jeweiligen Teilaufgaben ist. Teilaufgaben mit mehreren Lösungen oder mit widersprüchlichen Teilen werden mit 0 Punkten bewertet. *Make sure that it is absolutely clear what your final solution is for each sub question. Sub questions with multiple solutions or with contradicting parts are voided: 0 marks.*

Die folgende Tabelle wird von uns ergänzt! *The below table is completed by us!*

Aufgabe/Question	1	2	3	4	5	Total
Erreichbare Punkte <i>Possible marks</i>	12	12	12	12	12	60
Erreichte Punkte/ <i>Obtained marks</i>						
Note/Grade:						

Nachname/Last name	Vorname/First name	Matrikelnummer/ Matriculation number

Aufgabe/Question 1 (Zum Aufwärmen/Warm up, 3 + 3 + 1 + ... + 1 Punkte/marks)

1. „Zählen Sie drei prinzipielle Verfahren im Umgang mit Verklemmungen(*deadlocks*) auf und geben Sie zu jedem jeweils ein Beispiel an!“

“Enumerate three principal schemes for handling deadlocks and give a concrete example for each of these methods.”

a) **Prevention**

Resource allocation along a predefined resource type order.

b) **Avoidance**

Banker algorithm guaranteeing safe states

c) **Detection**

Observe system utilization (if too low, use Banker algorithm, and abort thread(s) if there is a deadlock)

2. „Zählen Sie mindesten drei typische TCP/IP Applikationen auf!“

“Enumerate at least three typical TCP/IP applications.”

a) **Simple Mail Transport Protocol (SMTP)**

b) **File Transfer Protocol (FTP)**

c) **Telnet**

Einige der folgenden Aussagen sind korrekt, einige inkorrekt. Unterstreichen Sie „korrekt“, wenn die Aussage korrekt ist, unterstreichen Sie „inkorrekt“, wenn die Aussage inkorrekt ist.

Some of the following statements are correct, some are incorrect. Underline “korrekt” if the statement is correct; underline “inkorrekt” if the statement is incorrect.

3. „RSA ist ein asymmetrisches Verschlüsselungsverfahren, das u.a. im Programmsystem PGP (*pretty good privacy*) verwendet wird.“

“RSA is an asymmetric encryption method used in the PGP-system (pretty good privacy).”

korrekt

inkorrekt

Fortsetzung von Aufgabe 1 / Question 1 continued: (1+1+1+1+1 Punkte/marks)

4. „Das Halbierungsverfahren (*buddy system*) verwaltet effizient den komplett ein-/auslagerbaren Hauptspeicherbereich (*main memory*) in einem virtuellen Speicher mit Seitenabbildung (*paging*).“

“*The buddy system efficiently manages the complete pageable part of main memory in a virtual memory system based on paging.*”

korrekt

inkorrekt

5. „Client-Server Applikationen mit „fetten Klienten“ (*fat clients*) erleichtern die Wartbarkeit (*maintainability*) eines verteilten Systems.“

“*Fat client applications enhance the maintainability of a distributed system.*”

korrekt

inkorrekt

6. „Ein wesentlicher Vorteil eines zustandslosen Dateidienstes (*fileserver*) –im Gegensatz zu einem zustandsbehafteten- ist seine verkürzte Zugriffszeit.“

“*One of the major advantages of a stateless file server – in contrast to a stateful server-- is its reduced access time.*”

korrekt

inkorrekt

7. „Das *Network File System (NFS)* der Firma Sun unterstützt nur Unix ähnliche Betriebssystemplattformen.“

“*Sun's network file system (NFS) supports only Unix like OS-platforms.*”

korrekt

inkorrekt

8. „Replikation in einem verteilten System ermöglicht neben geringerer Latenz vor allem eine bessere Verfügbarkeit (*availability*).“

“*Other than reduced latency, replication within a distributed system primarily improves availability.*”

korrekt

inkorrekt

Nachname/ <i>Last name</i>	Vorname/ <i>First name</i>	Matrikelnummer/ <i>Matriculation number</i>

Aufgabe/Question 2

(4 + 2 + 6 Punkte/marks)

1. „Beschreiben Sie die Eigenschaften der Lamportzeit und beschreiben Sie eine mögliche Implementierung der Lamportzeit!“

“Describe the properties of Lamport-time, and describe its potential implementation.”

Die Lamportzeit ist eine logische Zeit, bei der durch lokale Zähler die Kausalitätsrelation von relevanten Ereignissen (rein lokales, Sende- bzw. Empfangsereignis) in verteilten Systemen wie folgt realisiert wird:

Sei e ein lokales bzw. ein Sendeereignis. Hat e keinen lokalen Vorgänger, dann gilt:

$L(e) = 1$, ansonsten hat e einen Vorgänger e' , womit $L(e) = L(e') + 1$.

Sei e ein Empfangsereignis mit einem zugehörigen Sendeereignis s (im allg. auf einem anderen Knoten). Hat e keinen lokalen Vorgänger, dann gilt: $L(e) = L(s) + 1$, ansonsten hat e einen lokalen Vorgänger e' und es gilt: $L(e) = \max\{L(s), L(e')\} + 1$.

The Lamport-time is a logical time inducing some causal order of relevant events (local ones, sending, receiving) in a distributed system whereby the time is produced via local counters as follows:

Assume e is a local or a sending event. If there is no local predecessor of e , then $L(e) = 1$, otherwise there is at least one local predecessor e' , and thus: $L(e) = L(e') + 1$.

Assume e is a receiving event with a corresponding sending event s on some other node. If there is no local predecessor of e , then $L(e) = L(s) + 1$, otherwise there is at least one local predecessor e' , and then: $L(e) = \max\{L(s), L(e')\} + 1$.

2. „Die Lamportzeit erzeugt keine Totalordnung aller relevanten Ereignisse. Zeigen Sie dies durch ein kurzes Beispiel, und erweitern Sie die Lamportzeit so, daß dadurch eine Totalordnung erzeugt wird.“

“Lamport-time does not guarantee a total ordering of all relevant events. Show this fact in a small example, and extend the Lamport-time such that it provides total ordering.”

Zwei rein lokale, also unabhängige Ereignisse e und e' auf zwei verschiedenen Knoten können in jeder beliebigen Reihenfolge ausgeführt werden, d.h. sie sind nebenläufig $e \parallel e'$. Wenn es eine eindeutige Durchnumerierung der k involvierten Knoten gibt, dann kann diese Knotennummer hinzugezogen werden, um eine total geordnete Ereignismenge zu erhalten.

Two non-related (e.g. pure local) events e and e' on two different nodes can be executed in any order, i.e. e and e' are concurrent ($e \parallel e'$). If there is an unambiguous numbering of the k involved nodes, you can add this node number to achieve a total ordering between all relevant events.

Fortsetzung von Aufgabe 2 / Question 2 continued:

(6 Punkte/marks)

3. „Geben Sie eine knappe Beschreibung eines Algorithmus' Ihrer Wahl zur Verklemmungs-entdeckung (*deadlock detection*) in einem idealisierten, weil Hardwarefehlerfreien verteilten System (*distributed systems*) an. Diskutieren sie Vor- und Nachteile des gewählten Verfahrens!“

“Give a short description of an algorithm of your choice that implements deadlock detection in an idealized distributed system without any hardware failures. Discuss the pros and cons of your chosen algorithm!”

Variante 1:

Zentralisierter Algorithmus, in dem ein ausgezeichnete Knoten den Betriebsmittelzuordnungsgraphen (*resource allocation graph*) unterhält, der dann wie üblich auf Zyklensfreiheit untersucht werden kann. Hierzu muß jede Ressourcenanforderung bzw. -freigabe nicht nur dem entsprechenden Betriebsmittelverwalter (*resource management*), sondern eben auch diesem zentralen Knoten zugeschickt werden. Bei ungünstigen Laufzeiten zwischen Ressourcenfreigabe und neuer Ressourcenanforderungen kann es allerdings zu sogenannten Phantomverklemmungen kommen. Darüber hinaus sind pro Betriebsmittelvergabe bzw. -freigabe jeweils zwei Nachrichten notwendig.

Variante 2:

Bei jeder abgewiesenen Betriebsmittelanforderung an den zuständigen Betriebsmittelverwalter R_k durch einen Thread T_i , überprüft dieser im Falle eines belegten Betriebsmittels, ob der aktuelle Betriebsmittelbesitzer T_j wegen eines anderen Betriebsmittels wartet. Sofern dies nicht der Fall ist, bricht der Überprüfungsalgorithmus ab. Im anderen Fall wird dem entsprechenden Betriebsmittelverwalter eine um T_j verlängerte Überprüfungs- und Freigabemessage gesendet usw. Enthält die so produzierte Überprüfungs- und Freigabemessage $\langle T_i, T_j, \dots, T_i \rangle$, dann liegt ein zyklische Warten vor.

Die Auflösung der Verklemmung kann ineffizient werden, da man lokal nicht alle beteiligten Knoten kennt. Ferner könnte gleichzeitig auf mehreren Knoten der Entdeckungsalgorithmus angestoßen worden sein, was zu Zielkonflikten bei den zu terminierenden Threads bzw. Transaktionen führen könnte.

Nachname/Last name	Vorname/First name	Matrikelnummer/ Matriculation number

Aufgabe/Question 3

(4 + 4 + 2 Punkte/marks)

1. „Gegeben sei die folgende Programmskizze für das „Interaktionsproblem begrenzter Puffer“ (*bounded buffer problem*). Untersuchen Sie die Einsatzfähigkeit (korrekte Funktion, Auftreten von Verklemmungen, Fairneß etc.) dieser Lösung und bewerten Sie diese Lösung prinzipiell! “

“Given the following program draft to the bounded buffer problem. Analyze this proposal’s fitness for use (correct function, potential occurrence of deadlocks, fairness etc.) and grade it on principle!”

```

semaphore e = n;           // number of empty buffers
semaphore f = 0;           // number of filled buffers
semaphore mutex = 1;       // controlling adding to or
                           // taking from buffer

cobegin

producer: loop
    produce;
    P(b);P(e); add to buffer ; V(b); V(f);
end of loop

consumer: loop
    P(f);P(b); take from buffer; V(b);V(e);
    consume;
end of loop

coend

```

Das obige Programm enthält die Gefahr einer Verklemmung. Wenn der Produzent dreimal gefüllt hat und der Konsument mittlerweile P(f) passiert hat, wobei der Produzent jedesmal vor dem Konsumenten die P(b) Operation zuerst durchläuft. Dann hängt der Produzent an P(e) und der Konsument gleichzeitig an P(b).

Prinzipiell gilt, daß dieser Ansatz mit mehreren Semaphoren überdimensioniert und zudem ineffizient ist. Lösung mit Monitoren (siehe Folie 94 ist zumindest eleganter).

Fortsetzung von Aufgabe 3 / Question continued3: (4 + 2 Punkte/marks)

2. „Wir betrachten folgenden Algorithmus zum wechselseitigen Ausschluß kurzer kritischer Abschnitte, die keine Systemaufrufe enthalten: Bevor ein Thread in einen kritischen Abschnitt eintritt, ruft er *disable-interrupt* auf. Das bedeutet u.a., daß auch kein unterbrechungsinitiiertes Threadwechsel vorgenommen werden kann. Nach Verlassen des kritischen Abschnitts ruft der Thread *enable-interrupt* auf. Untersuchen Sie die Einsatzfähigkeit (korrekte Funktion, Auftreten von Verklemmungen, Fairneß etc.) dieses Algorithmus´ in Ein- und Mehrprozessorsystemen!“

“Consider the following algorithm for the mutual exclusion problem of short critical sections that contain no system calls: Each thread entering its critical section performs disable-interrupt, thus preventing any thread-switch initiated by an interrupt. After its critical section, a thread performs enable-interrupt. Analyze this algorithm’s suitability for use (correct function, potential occurrence of deadlocks, fairness etc.) in single- and multiprocessor systems.”

Einprozessorsysteme: Korrekte Funktion, der kritische Abschnitt wird zusammenhängend ausgeführt, da kein Threadwechsel möglich ist, somit kann auch kein zweiter Thread in den kritischen Abschnitt. Lösung ist zudem verklemmungsfrei und funktioniert für eine beliebige Anzahl von Threads. Sie ist auch fair, da nach Verlassen des kritischen Abschnitts alle anderen Threads die gleiche Chance haben, ihren kritischen Abschnitt zu betreten. Nachteilig wirkt sich jedoch aus, daß die Reaktionsfähigkeit des System auf Unterbrechungen beeinträchtigt wird. Ein Thread könnte dies dazu mißbrauchen, bevorzugt längere Zeit den Prozessor zu bekommen. Bei Programmierfehlern im kritischen Abschnitt kann es zum Systemstillstand kommen.

Mehrprozessorsysteme: Hier funktioniert obige Lösung nicht mehr, da das Aussperren von Unterbrechungen prozessorspezifisch ist, und sich somit gleichzeitig mehrere Threads im kritischen Abschnitt befinden können.

3. „Nehmen Sie ein Land mit Rechtsverkehr an. Ergänzen Sie dessen Straßenverkehrsordnung so, daß an einer gleichberechtigten Straßenkreuzung zumindest für KFZs mit vorne angebrachten Nummernschildern keine Verklemmung (*deadlock*) entsteht, solange jeder Fahrer sein Nummernschild kennt und gute Sichtverhältnisse herrschen.“

“Assume that in a country where the cars drive on the right hand side, and intersections exist where two roads cross with no signed “right of way”. Extend this country’s traffic law such that at such uncontrolled intersections there can be no deadlock, provided every driver entering the intersection can see every other driver’s registration plate, and each driver knows her/his own registration plate.”

Warten vier KFZs an einer gleichberechtigten Kreuzung, dann hat das KFZ mit dem kleinsten Autokennzeichenwert Vorfahrt.

Nachname/Last name	Vorname/First name	Matrikelnummer/ Matriculation number

Aufgabe/Question 4

(2 + 2 + 2 + 3 + 3 Punkte/marks)

1. „Manche Mikroprozessoren unterstützen verschiedene Seitengrößen, z.B. offeriert ein Pentiumprozessor die Standardgröße 4 KB und eine „Superseite“ à 4 MB. Geben Sie einen konkreten Grund an, warum solch große Seitengrößen angeboten werden!“

“Some microprocessors support different page sizes, e.g. a Pentium processor has a standard page size of 4 KB, and a ‘super page’ of 4 MB. Give a concrete reason why they offer such large pages.”

Für den Bildschirmpuffer (*frame buffer*) ist eine zusammenhängende Speicherung nützlich, da dessen Inhalt de facto immer ganz oder gar nicht gebraucht wird, somit wird nur ein einziger Eintrag im *Translation-Lookasidebuffer* für den ganzen Bildschirmpuffer verbraucht, was nicht nur das Nachladen reduziert, sondern auch mehr Platz für andere TLB-Einträge läßt. Beides zusammen kann zu einer deutlichen Leistungssteigerung beitragen.

2. „Geben Sie den Hauptgrund dafür an, warum man bei größeren logischen/virtuellen Adreßräumen zu mehrstufigen Seitentabellen übergeht!“

“Give the main reason why, with larger logical/virtual address spaces, we use multi-level page tables.”

Der Platzbedarf für eine einstufige Seitentabelle bei einem 64-bit großen logischen Adreßraum und einer Seitengöße von 4 KB übersteigt wohl auch noch zukünftige Rechner, selbst bei kleineren logischen Adreßraumgrößen versucht man den räumlichen Overhead für eine einstufige Seitentabelle zu vermeiden.

3. „Beschreiben Sie eine Realisierungsvariante einer invertierten Seiten-/Kacheltabelle und leiten Sie daraus deren Eintragsfelder (*entries*) für ein Mehrprogrammsystem ab!“

“Describe one implementation of an inverted page table, and deduce the contents of one of its entries for a multi-programming system.”

Zumindest die Task/Thread-ID und die virtuelle Seitennummer sollten enthalten sein. Bei einer globalen Ersetzungstrategie könnten auch noch ein Referenz- und ein Modifikationsbit nützlich sein.

Fortsetzung von Aufgabe/ Question 4 continued

(3 + 3 Punkte/marks)

4. „Beschreiben oder skizzieren Sie, wie Sie eintrittsinvariante Code-Seiten (*reentrant code pages*) über eine invertierte Seiten-/Kacheltabelle gemeinsam nutzbar machen können?!“
“*Describe or illustrate by a rough drawing how you would enable sharing reentrant code pages via an inverted page table.*”

Statt der Task/Thread-ID und der virtuellen Seitennummer enthält eine gemeinsame Kachel eine Kennung, daß sie eben eine gemeinsam benutzte Seite beherbergt und einen Verweis auf einen Deskriptor, in dem dann alle benutzenden Tasks mit ihren virtuellen Seitennummern verzeichnet sind.

5. „Manche Systeme implementieren gemeinsam genutzte Code- bzw. Datenseiten so, daß diese Seiten nicht verdrängt/ersetzt werden, solange mindestens noch ein(e) Thread/Task diese benötigt. Diskutieren Sie Vor- und Nachteile dieser Idee!“
“*Some systems implement shared code or data pages such that these pages are not swapped/replaced as long as one thread/task needs them. Discuss pros and cons of this approach.*”

Dieser Ansatz favorisiert Anwendungen, die aus mehreren Threads bestehen und die einen Großteil ihrer Interaktion über gemeinsame Seiten realisieren. Andererseits verführt er Anwenderprogrammierer dazu, ihre eigentlich *singlethreaded* Anwendungen mit einem zweiten *Dummythread* auszustatten, der alle „gemeinsam benutzten“ Datenseiten einmal anfaßt, so daß somit der Seitentausch ausgetrickst wird.

Nachname/Last name	Vorname/First name	Matrikelnummer/ Matriculation number

Aufgabe 5/ Question 5:

(6 + 3 +3 Punkte/marks)

1. Ein Angreifer (*attacker*) kann auf verschiedene Weisen ein Weitbereichsnetzwerk (WAN) beeinträchtigen. Zählen Sie mindestens drei verschiedene Angriffsarten auf das Netz auf und geben Sie jeweils ein Beispiel für eine entsprechende Schutzmaßnahme an!
An attacker can harm a wide area network (WAN) by various approaches. Enumerate at least three different types of attacks on a network, and give an appropriate protection mechanism to guard against each of those attacks.

Unterbrechung/Interruption

**Physischer Schutz der Leitungen bzw. redundante Leitungen/
physical protection of the lines or redundant lines**

**Abfangen bzw. Mithören/Interception or eavesdropping
Verschlüsselung/Encryption**

Abänderung/Modification

Verschlüsselung, Prüfsummen, Zeitstempel/Encryption, parity sums, time stamps

Einspielen/Fabrication

Authentisierung des Senders/authentication of the sender

2. „Was ist ein „*traffic padding encryption device*“ und gegen welche Angriffsart wird es verwendet?“
“What is a traffic padding encryption device, and against which type of attack is it used?”

**Gegen das schiere Mithören chiffrierter Botschaften auf einer Kommunikationsleitung, so daß aus der Kommunikationsintensität keine Rückschlüsse gezogen werden können
/against eaves-dropping of encrypted messages**

3. „Zählen Sie einige prozessorinterne bzw. prozessornahe Hardwareeinrichtungen auf, die als Basisschutzmechanismen in einem lokalen System verwendet werden!
“Enumerate some processor internal or nearby hardware facilities used as basic protection mechanisms within a local system.”

Prozessormodi /kernel/user mode in the CPU

Zugriffsbits in der Adreßtransformationseinheit/Accessbits in the MMU