

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

Schriftliche Prüfung Technische Grundlagen der Informatik IV, 2014-07-25

1

Schriftliche Prüfung Fachgebiet Telekommunikationsnetze

Technische Universität Berlin
 Fachgebiet Telekommunikationsnetze
 Prof. Dr.-Ing. Adam Wolisz

Technische Grundlagen der Informatik IV

2014-07-25

Dauer: 120 min

Name _____

Vorname _____

Matrikelnummer _____

Studiengang _____

Erklärung: (ohne Unterschrift wird die Klausur nicht korrigiert.)

Das in der vorliegenden Prüfung erzielte Ergebnis wird von mir als Fachprüfung bzw. Wiederholungsprüfung im Rahmen der für mich zutreffenden Prüfungsordnung auch dann als verbindlich anerkannt, wenn ich noch nicht in der von der Prüfungsordnung vorgesehenen Form angemeldet bin. Bei späterer Anmeldung wird die bei dieser Klausur erzielte Prüfungsnote übernommen.
 Ich bin damit einverstanden, dass meine Punktzahl und Note in anonymisierter Form auf ISIS veröffentlicht wird (ggf. streichen).

 Datum / Unterschrift

Hinweise:

- Schreiben Sie ihre **Matrikelnummer auf jeden Bogen** der Klausur, bevor Sie mit der Bearbeitung beginnen.
- Lesen sie die Aufgabenstellung genau!
- **Schreiben Sie bitte lesbar. Was wir nicht lesen können, gibt keine Punkte.**
- Wenn Sie zusätzliche, **von uns ausgegebene Blätter** oder die Rückseiten verwenden, geben Sie unbedingt an, zu welcher Aufgabe die Lösung gehört und versehen sie mit Matrikelnummer und durchgängiger Nummerierung.
- Es sind keine Hilfsmittel außer nicht-programmierbaren Taschenrechnern und Wörterbücher erlaubt.
- Halten Sie Ihre Antworten bitte so ausführlich wie nötig, aber **so kurz wie möglich**. Erklärungen sind nur nötig, wenn dies in der Aufgabenstellung explizit vermerkt ist.
- Bitte keinen Bleistift verwenden. Wer es doch tut, hat keine Möglichkeit mehr zur Beanstandung. Nur schwarze oder blaue Stifte verwenden.
- Es müssen nicht alle Punkte erreicht werden um eine 1,0 zu erhalten.

Ergebnis

Maximal mögliche Punktzahl	65
Zum Bestehen hinreichende Punktzahl	32,5
Erreichte Punktzahl	
Note	

1 Rechnernetze - Layer

- Physical-Layer:
 - verschlüsseln
 - duplizieren
 - transportiert die Bits

Bits
↑
↓

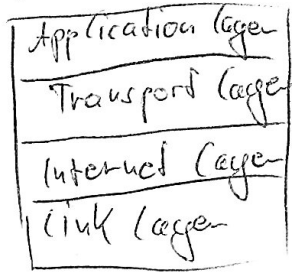
- Data-Link Layer:
 - Fehlererkennung
 - mehrfach Zugriff über broadcast channel
 - Adressierung

↑
↓

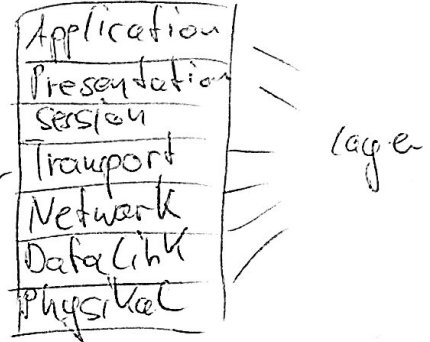
- Packets
- Unit s/s. 14 - mit Flusskontrolle:
 - zuverlässiger Daten transfer
 - transferiert frames von einem Knoten zu einem ihm benachbarten über einen Link (Wireless, wired, LANs)

↑
↓

Internet Model:



ISO/OSI:



- Network Layer:
 - ~~enabt.~~ Daten Transfer unabh. von benutzten Technologien in Schicht

transport zwischen
END SYSTEMS

- Verbindungslos und Verbindungsorientierte Services (UDP, TCP)
- handelt Netzwerkstaus/Verstopfung
- Routing (Pfadfindung von Paketen)

transport zwischen
Prozessen

- Transport Layer:
 - bietet logische Kommunikation zwischen application processes welche auf versch. Hosts laufen

Aufgabe 1:

Welche vier Arten von Verzögerungen können bei der Übertragung von Nachrichten auftreten? Bitte nennen und erläutern Sie diese **kurz**.

nodal processing : check bit errors
 Determining output
 (legt fest)

Queueing : Time waiting at output for transmission
 Depends on congestion at router
 (Verstopfung)

Transmission delay: $R = \text{link bandwidth (bps)}$
 $L = \text{packet length (bits)}$
 Time to send bits into link: $L/R \left[\frac{L}{R} \right]$

Propagation delay: $d = \text{length of physical link (Kabel/Länge)}$
 $s = \text{propagation speed in medium (Ausbreitungsgeschwindigkeit)}$
 $\text{propagation delay} = d/s \left[\frac{d}{s} \right]$
 Zeit die die Daten
 auf dem Kabel verbringt

	4
--	---

Aufgabe 2:

Gegeben sei ein Glasfaserkabel mit einer Länge von 200 km und einer Datenrate von 40 Gbit/s (1 Gbit = 10^9 Bits). Wieviele Bits befinden sich gleichzeitig auf der Leitung? Gehen Sie von einer Ausbreitungsgeschwindigkeit von 200 000 km/s aus. Geben Sie zusätzlich zum Ergebnis Ihren Rechenweg an!

$$\frac{200 \text{ km}}{200.000 \text{ km/s}} = 0,001 \text{ s}$$

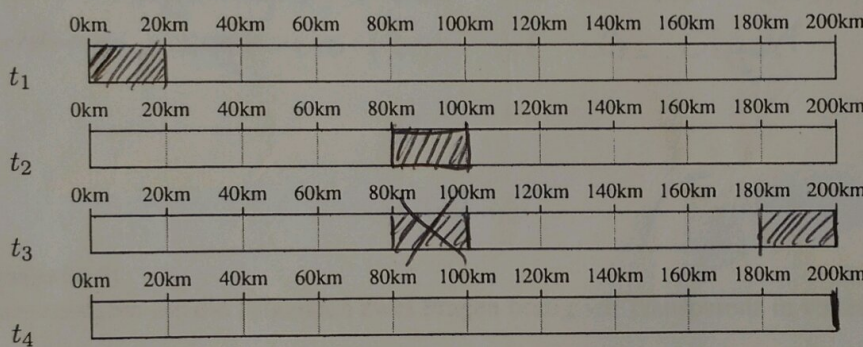
$$40 \text{ Gbit/s} \cdot 0,001 \text{ s} = 0,04 \text{ Gbit} = 40.000.000 \text{ Bit}$$

	2
--	---

Aufgabe 3:

Zwei Rechner seien durch eine Leitung von 200 km Länge miteinander verbunden. Vom Sender sollen Pakete mit einer Größe von 10 000 Bits bei einer Übertragungsrate von 100 Mbps (1 Mbps = 10^6 bits per second) übertragen werden. Processing und Queuing Delay seien vernachlässigbar klein. Zum Zeitpunkt $t_0 = 0 \mu s$ wird damit begonnen, ein einzelnes Paket zu senden. Die Ausbreitungsgeschwindigkeit der Signale beträgt $v = 200\,000 \text{ km/s}$.

- Wann beendet der Sender die Übertragung dieses Pakets (t_1)?
- Wann kommt das erste Bit am Empfänger an (t_3), wann das letzte (t_4)?
- Zeichnen Sie in das untenstehende Diagramm die Position des Paketes zu den von Ihnen berechneten Zeitpunkten sowie zum Zeitpunkt $t_2 = 500 \mu s$ ein, indem Sie den Bereich vom ersten bis zum letzten Bit des Pakets schraffieren. $500 \cdot 10^{-6} \text{ s}$
- Wann käme das letzte Bit am Empfänger an, wenn die Übertragungsrate 10 Mbps beträgt (t_{4b})? Was ergibt sich für eine Leitungslänge von 20 000 km und Übertragungsraten von 100 Mbps (t_{4c}) und 10 Mbps (t_{4d})?



$$t_1 = v \left[\frac{1}{10^6} \text{ s} = 200'000 \frac{\text{km}}{\text{s}} \right]$$

$$a.) \frac{10'000 \text{ Bit}}{100 \cdot 10^6 \frac{\text{Bit}}{\text{s}}} = \frac{1}{10'000} \text{ s} = t_1$$

$$b.) \frac{200 \text{ km}}{200'000 \frac{\text{km}}{\text{s}}} = \frac{1}{1000} \text{ s} = t_3 \quad t_3 + t_1 = t_4 =$$

$$c.) 500 \cdot 10^{-6} \text{ s} \cdot 200'000 \frac{\text{km}}{\text{s}} = 100 \text{ km}$$

$$d.) \frac{10'000 \text{ Bit}}{10 \cdot 10^6 \frac{\text{Bit}}{\text{s}}} = \frac{1}{1'000} \text{ s} \quad t_{4b} = \frac{1}{1'000} \text{ s} + \frac{1}{1000} \text{ s} = \frac{2}{1000} \text{ s} = \frac{1}{500} \text{ s} = t_{4b}$$

$$\cdot \frac{1}{10'000} \text{ s} + \frac{20000 \text{ km}}{200'000 \frac{\text{km}}{\text{s}}} = \frac{1}{10'000} \text{ s} + \frac{1}{10} \text{ s} = \frac{1001}{10'000} \text{ s} = t_{4c} \quad \boxed{6}$$

$$\cdot \frac{1}{1000} \text{ s} + \frac{1}{10} \text{ s} = \frac{1001}{1000} \text{ s} = t_{4d}$$

Aufgabe 4:

Nennen Sie vier der API-Funktionen der *Berkeley-Sockets* und erläutern sie diese jeweils kurz.

Socket - Create a new communication end point

Bind - Attach a local address to a socket

Listen - Announce willingness to accept connection

Connect ~~Accept~~ - Actively attempt to establish a connection

Accept ~~Connect~~ - Block caller until a connection request arrives

Receive - Receive some data over the connection

Close - Release the connection

	2
--	---

Aufgabe 5:

Greifen Sie die abstrakten Konzepte „Name“ und „Adresse“ voneinander ab.

Name sagt was | wen möchte man erreichen (Object Identity)

Adresse sagt wo | Wo ist das Objekt (Locator)

	1
--	---

Aufgabe 6:

Beantworten Sie die folgenden zwei Fragen bezüglich Transparenz in verteilten Systemen:

- Was versteht man im Zusammenhang mit verteilten Systemen unter Transparenz?
- Nennen und erläutern Sie in jeweils einem Satz drei Arten der Transparenz.

Unit 03

5.07 a) Es ist nicht von Interesse, dass ein vert. System ein Zusammenschluss von verschiedenen kleinen, einfacheren Systemen ist. Der Nutzer merkt davon nichts.

5.15

b) Access-Hide - Versteckt wie auf eine Ressource zugegriffen wird

Location - Versteckt wo die Ressource ist.

Migration - Versteckt, falls die Ressource verschoben wird

Relocation - ~~ist~~

während sie benutzt wird

	4
--	---

Replication - Versteckt das Kopieren von Ress.

Aufgabe 7:

- a) Erläutern sie die Vorgehensweise beim Speichern/Lesen eines Wertes in einer *distributed hash table (DHT)*.
- b) Wie funktioniert die *Chord Joining Operation*?

a) ?

b) Unit 05a S. 11

- jeder Node A sendet stabilize an Nachfolger B

	3
--	---
 - Wenn B stabilize erhält, sendet er seinen Vorgänger zu A zurück.
 - A erhält Vorgänger von B, wenn nicht A selbst update von A Nachfolger
-

Finger-table:

Aufgabe 8:

Das Domain Name System (DNS) hat die Hauptaufgabe der Beantwortung von Anfragen zur Namensauflösung.

- a) Erläutern Sie den Unterschied zwischen iterativer und rekursiver Namensauflösung.
- b) Viele DNS-Host-Namen können zu mehreren IP-Adressen aufgelöst werden. Nennen Sie einen Grund, warum dies sinnvoll sein könnte?
- c) Der autoritative Nameserver bestimmt für die DNS-Einträge einen TTL-Wert (engl.: Time to live; dt.: Zeit zu leben). Was bedeutet dieser Wert und was ist der Vorteil von hohen Werten?

Unit 06 S. 12/13
a) rekursiv: Die Anfrage wird solange weitergereicht, bis eine die Antwort hat, diese wird über den gleichen Weg wieder zurückgeschickt
iterativ: Ich frage, der kennt die Antwort nicht, sagt mir aber wer sie kennen könnte, solange bis Antwort erhalten

b) z.B. um einen leeren oder näheren Server zu finden. auch Folien
z.B. Google-Server in FFN nicht Amiland

S. 15 c) In Warkschlange geachte Antworten haben einen TTL-Wert. Ist die Zeitspanne abgelaufen, so wird die Antwort aus dem Cache gelöscht.
Was länger im Cache bleibt muss nicht so oft nachgeguht werden, bzw. kann schneller geladen werden

5

Aufgabe 9:

Verwendung von Uhren in verteilten Systemen:

- a) Nennen Sie ein Problem, welches auftreten kann, wenn in verteilten Systemen lokale Uhren verwendet werden?
- b) Nennen Sie zwei Ansätze für die Lösung des in a) genannten Problems.

Unit 06
S. 6 a) Wenn jede Maschine seine eigene Uhrzeit hat, dann kann es passieren, dass ein Event, welches nach einem anderen aufgetreten ist, dennoch eine frühere Zeit zugewiesen bekommt.

-The file has been compiled before its creation???

S. 8 b) -Periodic Clock sync. (löst eine Uhr z.B. (ausgesam) (aufen))

3

- Logical Time: order of events is often sufficient (ausreichend)
[Reihenfolge]

Uhren:

Christian Algo: $P :=$ Prozess $S =$ Zeitserver

- P fragt S nach Zeit zum Zeitpunkt t_0
 - S verarbeitet die Frage (benötigt Zeit J)
 - Die Antwort $C_M(t_1)$ wird von P zur Zeit t_1 empfangen
 - P setzt seine Uhr auf $C_M(t_1) + \frac{RTT}{2}$
(Die Zeit vom Server + Round trip time)
- $RTT = (t_1 - t_0)$ - oder - besser $RTT = (t_1 - t_0 - J)$

Berkeley Algo:

1. Zeitserver fragt regelmäßig Zeiten von Clients ab.
2. Clients antworten (schicken Ihre Zeit - server Zeit) (schickt seine Zeit mit)
3. Zeitserver rechnet AVG aus allen Zeiten aus (ermittlung über Christian)
4. schickt allen Client die Differenz die sie ändern müssen (setzt sich selbst auch neu (maybe))
justiert

Veektor Uhr: • Zuweisung von eindeutigen Timestamps

Logische Uhr

- erlaubt Nebenläufigkeit von Ereignissen zu ermitteln
- Fortführung von Lamport Uhren

(NTP):

- + Fehler tolerant
- + Unabhängig von time zones
- + Highly scalable

→ mehrere Anfragen \Rightarrow minimiere Fehler (Wählt beste Quelle)

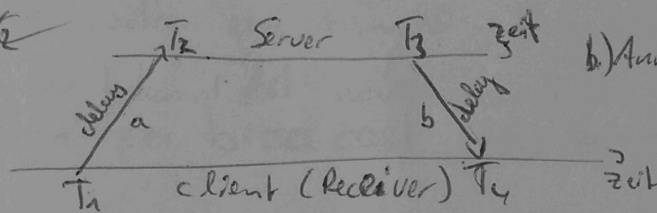
→ referenz Uhren Quellen

- sub nodes sind in einer Hierarchie

Aufgabe 10:

- a) Leiten Sie (unterstützt durch eine Skizze!) die Berechnung des Offsets zwischen einem Client und einem Server her, die das Network Time Protocol (NTP) verwenden.
- b) Welche vereinfachende Annahme machen Sie bei der Berechnung in Teil a)?

Unit 06

S. 19 a) Offset: $\Theta = \frac{1}{2} \Delta t$ 

b.) Annahme: a und b sind gleich lang

damit Delay als Offset raus

$$\text{Offset} = \Theta = \frac{1}{2} [(T_2 - T_1) + (T_3 - T_4)]$$

$$\text{Delay} = \delta = (T_4 - T_1) - (T_3 - T_2)$$

- Offset: Unterschied zwischen eigener Zeit und Referenzzeit
- Delay: Zeit von Anfrage, bis Eintreffen der Antwort

	3
--	---

~~XXXXXXXXXX~~

$$\delta: \text{Delay} = \delta = (t_4 - t_1) - (t_3 - t_2) = \delta$$

$$\Theta: \text{Offset} = \Theta: t_2 = t_1 + \delta \cdot \frac{1}{2} + \Theta$$

$$\Theta = t_2 - t_1 - \delta \cdot \frac{1}{2}$$

$$\Theta = t_2 - t_1 - \frac{1}{2} ((t_4 - t_1) - (t_3 - t_2))$$

$$\Theta = \underbrace{t_2 - t_1}_{\text{min}} - \frac{1}{2} t_4 + \frac{1}{2} t_1 + \frac{1}{2} t_3 - \frac{1}{2} t_2$$

$$\Theta = \frac{1}{2} t_2 - \frac{1}{2} t_1 - \frac{1}{2} t_4 + \frac{1}{2} t_3$$

$$\Theta = \frac{1}{2} ((t_2 - t_1) + (t_3 - t_4))$$

~~XXXXXXXXXX~~

PTP:

- + : 20-100 x mehr Genauigkeit als NTP
- : Problems under heavy load

Aufgabe 11:

In einem Verteilten System fällt der Koordinator aus. Beschreiben Sie die einzelnen Schritte des Bully-Algorithmus zur verteilten Wahl eines neuen Koordinators.

Unit 08 S. 31-33

Prozess bemerkt, das Koordinator tot, startet eine neue Auswahl, indem er an alle höheren IDs eine Nachricht schickt. Diese senden eine Antwort und starten selber eine Auswahl. Der Prozess, der keine Antwort mehr bekommt ist neuer Koordinator. Da höchste ID. Er schickt allen per broadcast, dass er der neue Koord. ist.

	4
--	---

Aufgabe 12:

- Erläutern Sie kurz das *Primary-Copy* Verfahren zur Synchronisation replizierter Datenbanken.
- Was ist der wesentliche Vorteil gegenüber *Read-Once-Write-All (ROWA)*?
- Welchen Nachteil hat das Verfahren?

Unit 07 S. 34-40

ROWA: Auf allen nodes sind alle Daten synchronisiert und aktuell.
 +: ~~Ausfall~~ von nodes leicht zu kompensieren Daten sind auf allen Knoten vorh.
 -: Aktualisierung erfordert lock und synchronize aller nodes
 (-: Ausfall von node benötigt Berücksichtigung aller anderen nodes mit gleichen Datensatz) Major

Primary-Copy: Eine Master-Datei wird als primary-copy gesetzt, diese wird sofort geupdated.
 Andere dupliziert data sets werden asynchron vom primary geupdated.

	3
--	---

- + : Ausfall von nicht primary-nodes leicht zu handeln
- : verzögerte ~~data modification von duplizierten data sets~~ Modifikation von duplizierenden data sets.

Aufgabe 13:

Beschreiben Sie die vier Schritte des *Two-Phase Commit* Protokolls zwischen einem Koordinator und N Teilnehmern. Dabei sei die Kommunikation fehlerfrei. Es soll aber beachtet werden, dass Teilnehmer sich sowohl für als auch gegen einen commit entscheiden können.

[2 PC]

OK!
 1. Koord: can commit? (Beteiligte) andere: yes we can!
 Koord: do commit!
 andere: have comid!
 unsicher

FAIL:
 2. Koord: can commit?
 mind 1.: no!
 Koord: abort (alles rückgängig)
 andere: have undone
 •Vote reject
 •Vote
 •Decision
 •Acknowledge

3 Phase Commit: 3PC

1. Koord: canCommit?
 Beteiligte: yes we can! — hier zwischen ist
 Koord: pre Commit! unsicher
 Beteiligte: pre Commit Ack
 Koord: Commit (do)
 Beteiligte: have commit

5

Aufgabe 14:

Die Bitfolge 01101001 soll vor der Übertragung mit einer Prüfsumme versehen werden. Dazu soll CRC (zyklische Redundanzprüfung, englisch: cyclic redundancy check) verwendet werden. Benutzen Sie als Generator-Polynom $x^4 + x^3 + x^0$. Wie lautet die komplette Bifolge, die übertragen werden muss?

grad 5 \Rightarrow 5-1 Nullen davor (an Rahmen)

genpoly: 11001 \Rightarrow 110010000
 Rahmen: 01101001 + 0000 \leftarrow Anhang
 110100110000
 11001
 00011010
 11001
 00011000
 11001
 00001 \rightarrow Rest

3

Rest (4 Stellen)
 Bitfolge Rest
 to send: 110100110001 - checksumme
 PROBE:
 11001
 00011010
 11001
 00011001
 11001
 0



Aufgabe 15:

- a) Was ist das Address Resolution Protocol (ARP) und wofür wird es benutzt? Unit 09
S. 41
 b) Beschreiben Sie den Ablauf eines ARP-Request.

a.) Es löst ein vorhandene IP zu einer gesuchten MAC Adresse auf.
 Es ist eine Tabelle in ~~allen~~ ^{jeden} IP Knoten ~~des~~ eines Netzwerkes

b.) A möchte B Daten senden, keine MAC in ARP-table
 • A broadcastet ARP Paket mit B's IP.
 • B erhält ARP Paket mit seiner IP und unicast ~~zu~~ ~~A's~~ ~~Mac~~ zu A's Mac ein Paket mit B's Mac
 • A speichert das paar (B IP & Mac) in A's ARP-table bis TTL abgelaufen.

	3
--	---

Aufgabe 16:

Wie funktioniert Classless Inter-Domain Routing (CIDR) und was sind die Vorteile?

Addressbereiche können in 2er Potenzen zugeteilt werden

Addressformat a.b.c.d/x ~~x ist subnetz Anteil~~

Bsp. a.b.c.d./28 => 1...1 0000 $\leftarrow 32-28=4$
 \downarrow 28x Möglichkeit für Host, also 2^4 versch. Hosts möglich
 #1 in Netzmaske

- + Es ist nicht auf A, B, C, D ~~W~~ Typ festgelegt
- + Je nachdem ob man viele Netzwerke und wenig Hosts oder umgekehrt braucht, geht es
- + Reduziert Größe von routing-Tabellen, aber bewahrt connectivity.

	2
--	---

Aufgabe 17:

- a) Erläutern Sie kurz das Konzept der Network Address Translation (NAT).
 b) Nennen Sie zwei sinnvolle Gründe, NAT zu verwenden.

a.) Nach Außen hat ein Netzwerk nur 1 IP, aber Innen $\hat{=}$ kann es mehrere geben (mehrere private IP, eine öffentlich)

b.) Knappheit von IPv4 Adressen (öffentlich)
 • wird als Trennung von internen und externen Netzen eingesetzt

	3
--	---

Unit 11 S. 16

Aufgabe 18:

Welche drei unerwünschten Ereignisse können Paketen bei einem Network-Layer, der nur *Best Effort* garantiert, widerfahren? Nennen Sie bitte zu jedem der Ereignisse einen Ansatz, mit dem dennoch ein zuverlässiger Transport gewährleistet werden kann!

- Paket können corrupted werden: wird mit CRC erkannt, Lsg. neu senden
- können verloren gehen: Lsg. Acknowledge + Timeout, Lsg. neu senden
- Reihenfolge falsch: Sequenznummern, Lsg. Empfänger buffert um neu zu ordnen

6

Aufgabe 19:

Zwei Stationen kommunizieren über ein *Go-Back-N-Protokoll* mit der Fenstergröße N (maximale Anzahl sich gleichzeitig auf dem Weg befindende unbestätigte Pakete).

Unit 11 S. 32

- Wie groß ist die erforderliche Puffergröße auf Sender- und Empfängerseite?
- Bei fehleranfälligen Kanälen kann es sinnvoll sein, nur wirklich verlorene Pakete nochmals senden zu lassen. Womit bezahlt man dies?

Selective Repeat
Alternative zu GBN

- N
- .

3