



Technische  
Universität  
Braunschweig

Institut für Betriebssysteme  
und Rechnerverbund  
Connected and Mobile Systems



# Computernetze 1

Fragestunde

Fynn Schulze, 08. August 2024

Technische Universität Braunschweig, IBR

# Überblick

## ■ Fragen

- Subnetze
- Routing-Tabellen
- Fehlererkennung und -korrektur
- Distance Vector Routing
- Sliding Window
- Ausbreitungs- und Übertragungsverzögerung
- Allgemeine Fragen

## ■ Organisatorisches



## Übung 8: 2b) Subnetze

- b) Eine Firma bekommt einen neuen Netzbereich mit den IP-Adressen im Netzbereich 134.169.0.0 - 134.169.255.255 (134.169.0.0/16). Die Firma besteht aus drei Abteilungen mit etwa gleich vielen Nutzern. Das Netz soll somit in drei möglichst gleich große Netze geteilt werden, die über einen Router verbunden sind.  
(Originalaufgabe hatte /24, hier /16)
- Welche IP-Nummern erhalten die jeweiligen Netze?
  - Wieviele Hosts können angeschlossen werden?
  - Wie lauten die Broadcast-Adressen?
  - Wie lauten die Netzwerk-Adressen?



## 2 b) Subnetze

i. Welche IP-Nummern erhalten die jeweiligen Netze?

### Lösung

- Netz lässt sich nur in 2er-Potenzen teilen!
- Variante A: gleich große Netze
  - 2 Bits für 3 Netze
  - $2^2 = 4$  (gleich große) Teilnetze
- Variante B: drei Netze
  - Teilen des Netzes in 2 Teile
  - Danach erneute Teilung eines Netzes
  - 3 Teilnetze von unterschiedlicher Größe



## 2 b) Variante A – Netzadressen

### Netzmasken der Variante A

- Für alle Teilnetze gleich → erste 18 Bit auf 1 gesetzt
  - 11111111.11111111.11000000.00000000 (255.255.192.0)

### Teilnetze der Variante A

- 1: 10000110.10101001.**00**000000.00000000 → 134.169.0.0/18
- 2: 10000110.10101001.**01**000000.00000000 → 134.169.64.0/18
- 3: 10000110.10101001.**10**000000.00000000 → 134.169.128.0/18
- 4: 10000110.10101001.**11**000000.00000000 → 134.169.192.0/18



## 2 b) Variante B – Netzadressen

### Netzmasken der Variante B

- Unterschiedlich je nach Teilnetz:
  - 1: 11111111.11111111.10000000.00000000 (255.255.128.0)
  - 2: 11111111.11111111.10000000.00000000 (255.255.128.0)
    - 2a: 11111111.11111111.11000000.00000000 (255.255.192.0)
    - 2b: 11111111.11111111.11000000.00000000 (255.255.192.0)

### Teilnetze der Variante B

- 1: 10000110.10101001.00000000.00000000 → 134.169.0.0/17
- 2: 10000110.10101001.10000000.00000000 → 134.169.128.0/17
  - 2a: 10000110.10101001.10000000.00000000 → 134.169.128.0/18
  - 2b: 10000110.10101001.11000000.00000000 → 134.169.192.0/18

## 2 b) Anzahl Host-Adressen

ii. Wieviele Hosts können angeschlossen werden?

### Lösung

⇒ Anzahl Host-Adressen:

- Variante A: 14 Bit Host-Anteil (4 Netze)
- Variante B: 14 Bit Host-Anteil (2 Netze), 15 Bit Host-Anteil (1 Netz)
- Broadcast-Adresse: alle Host-Bits 1
- Netzadresse: alle Host-Bits 0
- Variante A:  $4 \cdot (2^{14} - 2) = 65528$
- Variante B:  $2 \cdot (2^{14} - 2) + 1 \cdot (2^{15} - 2) = 65530$



## 2 b) Broadcast-Adressen

iii. Wie lauten die Broadcast-Adressen?

### Lösung

Variante A:

- Teilnetz 1: 134.169.63.255
- Teilnetz 2: 134.169.127.255
- Teilnetz 3: 134.169.191.255
- Teilnetz 4: 134.169.255.255

Variante B:

- Teilnetz 1: 134.169.127.255
- Teilnetz 2a: 134.169.191.255
- Teilnetz 2b: 134.169.255.255



## 2 b) Netzwerk-Adressen

iv. Wie lauten die Netzwerk-Adressen?

### Lösung

Variante A:

- Teilnetz 1: 134.169.0.0
- Teilnetz 2: 134.169.64.0
- Teilnetz 3: 134.169.128.0
- Teilnetz 4: 134.169.192.0

Variante B:

- Teilnetz 1: 134.169.0.0
- Teilnetz 2a: 134.169.128.0
- Teilnetz 2b: 134.169.192.0



# Adressierung

- Wie kommt man auf die höchste Host-Adresse?

## Lösung

- Netzwerk: 134.169.34.0/24
- Reservierte Adressen:
  - Netz-Adresse: 134.169.34.0 (kleinste Adresse)
  - Broadcast-Adresse: 134.169.34.255 (größte Adresse)
- Host-Adressen: 134.169.34.1 - 134.169.34.254
- Niedrigste Host-Adresse = Netz-Adresse + 1
- Höchste Host-Adresse = Broadcast-Adresse - 1



## Übung 8: 3b) Aufspaltung der Routing-Tabelle

Frage: Bitte nochmal erklären!

- b) Die IP-Adressen von 134.169.0.0 bis 134.169.127.255 wurden zu 134.169.0.0/17 zusammengefasst. Es stellt sich jedoch heraus, dass ein Bereich von 1024 Adressen (134.169.60.0 - 134.169.63.255) nun über eine andere Ausgangsleitung geroutet werden soll. Müssen die zusammengefassten Adressbereiche wieder aufgespalten werden oder gibt es noch eine andere Möglichkeit?

### Lösung

- Aufspalten ist **nicht** nötig
- Hinzufügen von 134.169.60.0/22 reicht
- Bereiche dürfen überlappen!
- "Der spezifischere gewinnt"

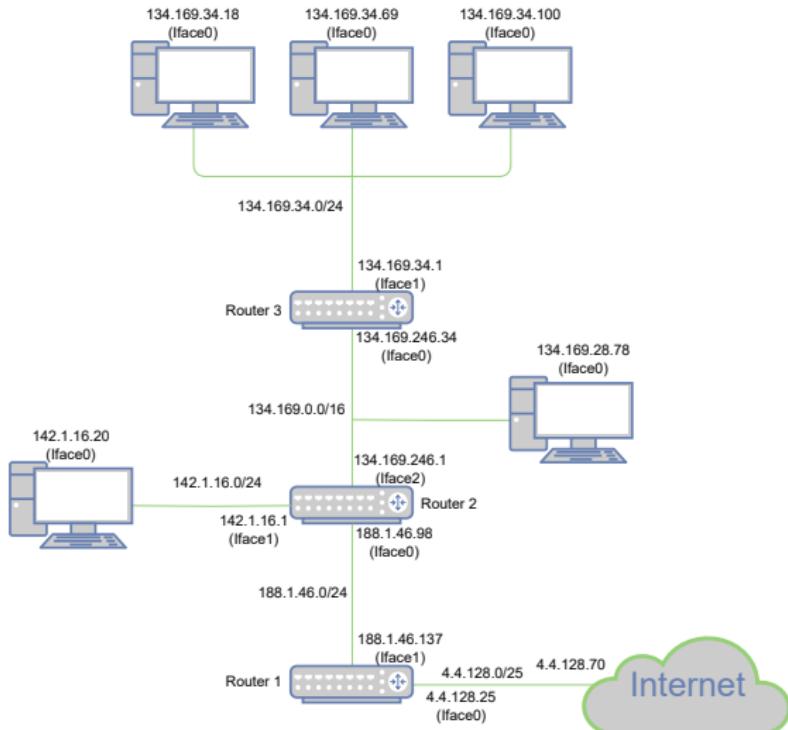


# Übung 9: 1a) Routing-Tabellen

1. Wie geht man beim ausfüllen der Routingtabelle vor?
2. Wann muss welches Ziel nicht aufgeschrieben werden?
  - Direkt angeschlossene Netze müssen angegeben werden
  - Default Route (meist ins Internet) muss angegeben werden
  - Sonst nur spezifische Ziele
    - z.B. andere Netze, die nicht direkt angeschlossen sind und nicht über die Default Route erreicht werden
  - Routen sollten nicht angegeben werden, wenn sie redundant sind
    - z.B. Subnetze von bestehender Route, wenn Gateway & Interface gleich



# Übung 9: 1a) Routing-Tabellen



# Übung 9: 1a) Router 1

## Routingtabelle Router 1

Ziel	Gateway	Netzmaske	Interface
134.169.34.0	188.1.46.98	255.255.255.0	1
134.169.0.0	188.1.46.98	255.255.0.0	1
188.1.46.0	*	255.255.255.0	1
142.1.16.0	188.1.46.98	255.255.255.0	1
4.4.128.0	*	255.255.255.128	0
default	4.4.128.70	0.0.0.0	0



# Übung 9: 1a) Router 2 & 3

## Routingtabelle Router 2

Ziel	Gateway	Netzmaske	Interface
134.169.0.0	*	255.255.0.0	2
134.169.34.0	134.169.246.34	255.255.255.0	2
142.1.16.0	*	255.255.255.0	1
188.1.46.0	*	255.255.255.0	0
default	188.1.46.137	0.0.0.0	0

## Routingtabelle Router 3

Ziel	Gateway	Netzmaske	Interface
134.169.34.0	*	255.255.255.0	1
134.169.0.0	*	255.255.0.0	0
default	134.169.246.1	0.0.0.0	0



# Weitere Fragen

- Gibt es beim Erstellen der Routingtabellen eine festgelegte Reihenfolge (Router 1, 2, 3, usw.), oder ist man in der Klausur frei was die Reihenfolge angeht?  
⇒ In diesem Fall gibt es keine feste Reihenfolge. Wenn es eine gibt, wird das explizit angegeben.
- Ist das Kürzen der Routingtabellen klausurrelevant?  
⇒ Je nach Aufgabenstellung, also ja.



# (Klassisches) Paritätsbit

Nutzen: Fehlererkennung bei 1-bit Fehlern

Beispiel: Paritätsbit für geraden Paritätscheck (Anzahl Einsen immer gerade)

ursprüngliches Wort	+ Paritätsbit	= resultierendes Wort
00	0	000
01	1	011
10	1	101
11	0	110

- Wie setzt man Paritätsbits richtig?
  - Kommt auf den verwendeten Code an
  - Gibt sehr viele verschiedene Möglichkeiten
  - Für Klausur ist nur das Konzept wichtig
- Wie tragen Paritätsbits zur Fehlererkennung und -korrektur bei?

# Hamming-Distanz

Nutzen: Gibt Garantien zu gegebener Codierung, wie viele Bitfehler erkannt bzw. korrigiert werden können.

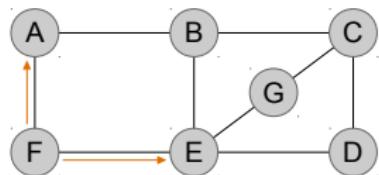
A	00000
B	10011
C	01110

- Distanzen:  $AB = 3$ ,  $AC = 3$ ,  $BC = 4$
- Hamming-Distanz: 3 (Minimum der Distanzen)

	Allgemein	Hier
Erkennung	$d \geq f + 1$	$2 \geq f$
Korrektur	$d \geq 2 \cdot f + 1$	$1 \geq f$



# Übung 7: 2a) Distance Vector Routing



Ziel	A	B	C	D	E	F	G
A	0	1	(2,B)	(3,C)	(2,B)	1	(3,C)
B	1	0	1	(2,C)	1	(2,A)	(2,C)
C	(2,B)	1	0	1	(2,B)	(3,E)	1
D	(3,F)	(2,C)	1	0	1	(2,E)	(2,C)
E	(2,B)	1	(2,B)	1	0	1	1
F	1	(2,A)	(3,B)	(2,E)	1	0	(2,E)
G	(3,F)	(2,C)	1	(2,C)	1	(2,E)	0

- F sendet seinen Vektor an A und E: [1, 2, 3, 2, 1, 0, 2]

⇒ A lernt, D und G über F zu erreichen

Frage: Man kommt von A zu D in 3 Hops über F. Ging das auch über B?

Antwort: Theoretisch ja, aber da wir alphabetisch vorgehen, bekommt A die Route zu D über F vorher mitgeteilt. Daher wäre es in diesem Fall falsch.

## Frage:

Müssen wir beim Distance Vector Routing alle Schritte bis ins Detail angeben oder würde es reichen, wenn man die Tabelle richtig ausfüllt?  
Alle Schritte anzugeben, würde viel Zeit kosten in der Klausur.

## Antwort:

Kommt auf die Aufgabenstellung an. Wenn danach gefragt wird, dann ja.  
Je aufwändiger die Aufgabe, desto mehr Punkte gibt es dafür.



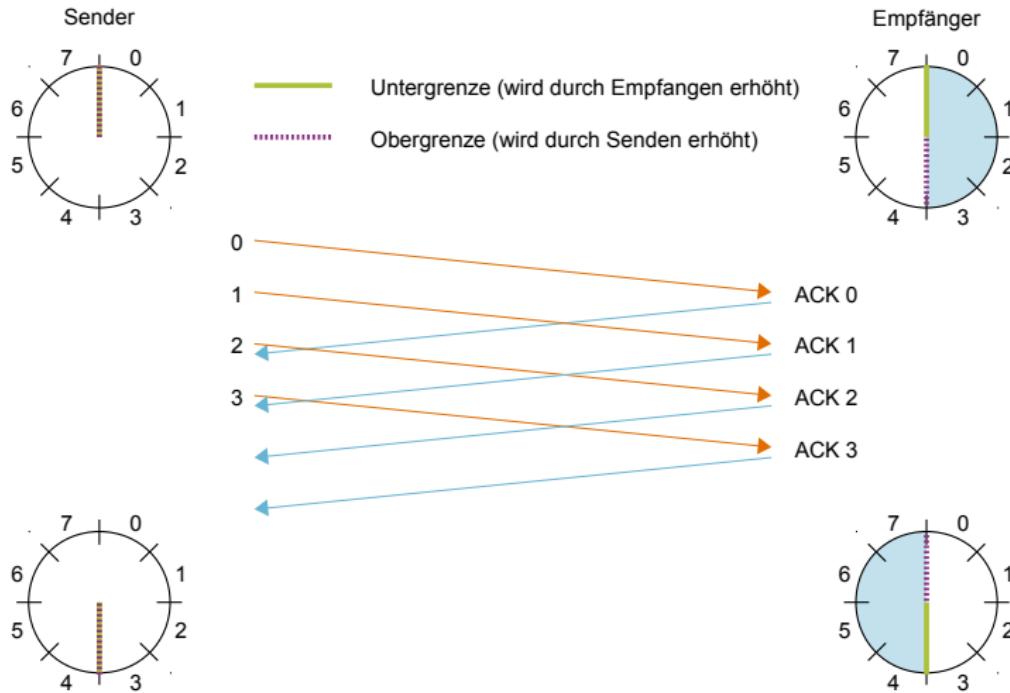
# Übung 5: 3) Sliding Window

## Erklärung

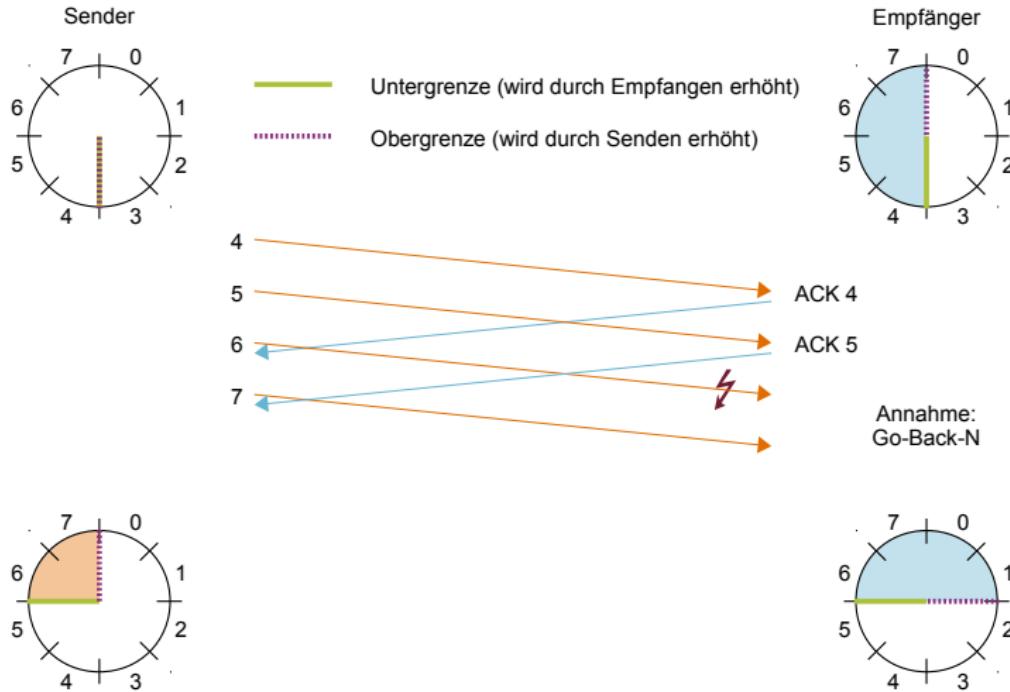
- Sequenznummernraum = 8 (0, 1, 2, 3, 4, 5, 6, 7)
- Fenstergröße = 4
- Datenframes werden positiv bestätigt  $\Rightarrow$  ACKs, wenn richtig empfangen
- nicht kumulativ  $\Rightarrow$  jeder Frame wird einzeln bestätigt
- Uhren markieren die gerade verwendeten Sequenznummern



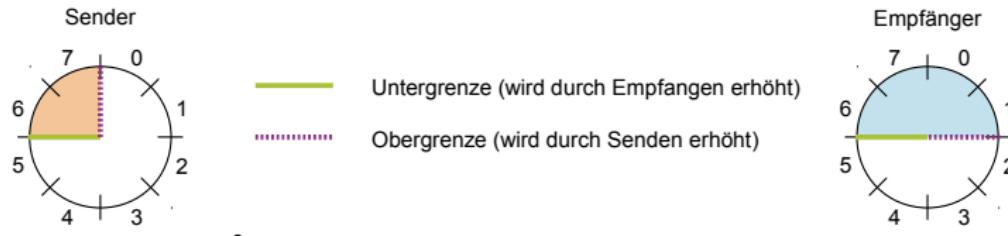
# Sliding Window: Schritt 1



# Sliding Window: Schritt 2



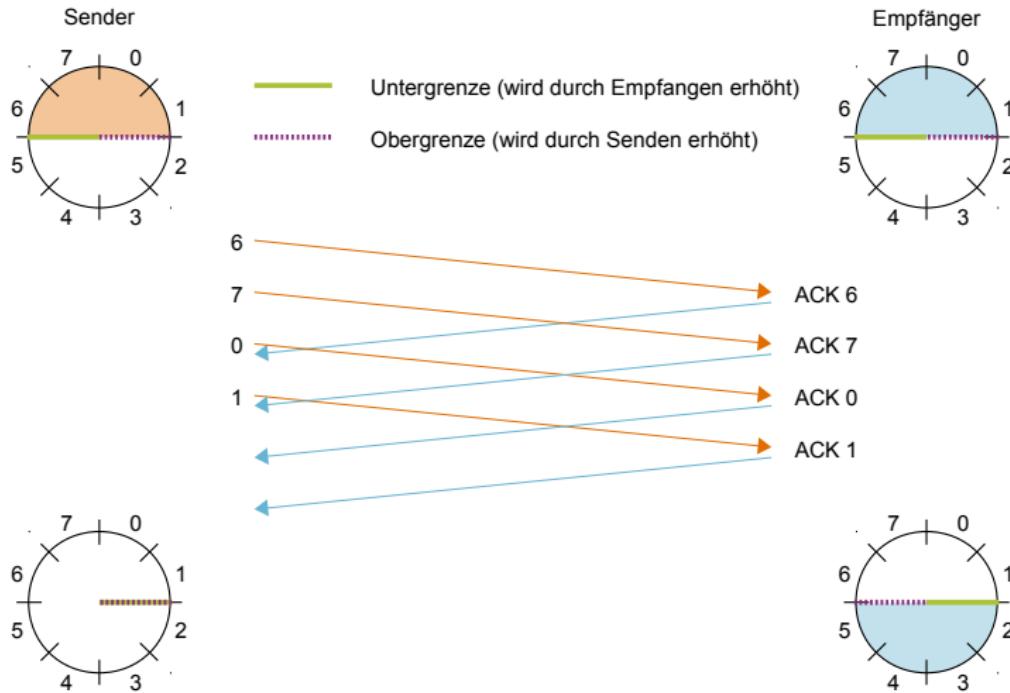
# Sliding Window: Schritt 3



Timeout



# Sliding Window: Bonus – möglicher Schritt 4

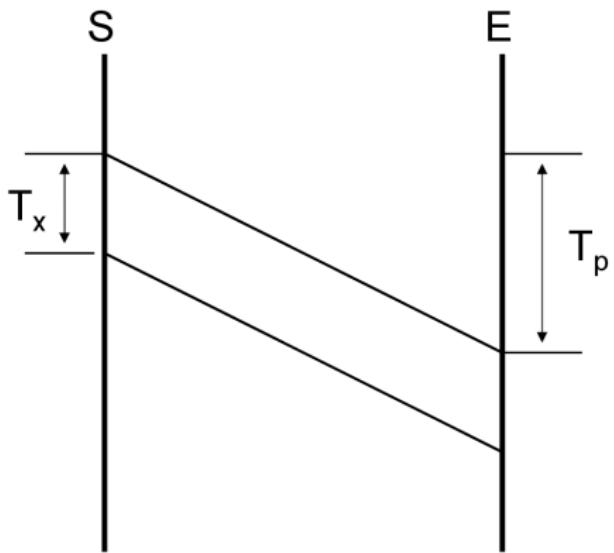


# Übung 3: 9) Verzögerungen

Signale breiten sich im luftleeren Raum mit Lichtgeschwindigkeit (etwa  $3 \cdot 10^8 \frac{m}{s}$ ) aus. In elektrischen Leitern (verdrillte Kupferkabel, Koaxialkabel) beträgt die Ausbreitungsgeschwindigkeit etwa  $2 \cdot 10^8 \frac{m}{s}$ . Berechnen Sie  $T_p$  und  $T_x$  für die Übertragung eines 1.000 Bit Blocks über 50 m verdrilltes Kupferkabel mit einer Bitrate von 10 kbit/s, Geben Sie bei allen Berechnungen die Zwischenschritte an und führen Sie die Einheiten bei der Berechnung mit! Notieren Sie alle Ergebnisse in Sekunden.



# Verzögerungen



## Definitionen

$T_p$  **Ausbreitungsverzögerung**  
(propagation delay): Abstand von Sender und Empfänger geteilt durch die Ausbreitungsgeschwindigkeit des Mediums

$T_x$  **Übertragungsverzögerung**  
(transmission delay): Anzahl der übertragenen Bits geteilt durch die auf dem Medium realisierte Bitrate



# Übung 3: 9a) Kupferkabel

a) 50 m verdrilltes Kupferkabel mit einer Bitrate von 10 kbit/s

## Ausbreitungsverzögerung $T_p$

$$d = 50 \text{ m}$$

$$v = 2 \cdot 10^8 \frac{\text{m}}{\text{s}}$$

$$\begin{aligned} T_p &= \frac{d}{v} \\ &= \frac{50 \text{ m}}{2 \cdot 10^8 \frac{\text{m}}{\text{s}}} \\ &= \frac{5 \cdot 10^1 \text{ m} \cdot \text{s}}{2 \cdot 10^8 \text{ m}} \\ &= 2.5 \cdot 10^{-7} \text{ s} \end{aligned}$$

## Übertragungsverzögerung $T_x$

$$n = 1000 \text{ bit}$$

$$s = 10000 \frac{\text{bit}}{\text{s}}$$

$$\begin{aligned} T_x &= \frac{n}{s} \\ &= \frac{1000 \text{ bit}}{10000 \frac{\text{bit}}{\text{s}}} \\ &= \frac{1}{10 \frac{1}{\text{s}}} \\ &= 0.1 \text{ s} \end{aligned}$$



# Allgemeine Fragen

## Frage:

Es hieß: (theoretische) Fragen aus der Vorlesung und (praktische) Aufgaben aus der Übung. In der Übung gibt es ja aber auch theoretische Fragen. Zählt das zu den *Theoriefragen aus der Vorlesung* oder zum *Übungswissen*?

## Antwort:

Theorie bleibt Theorie und Praxis bleibt Praxis.

Ist nicht weiter relevant, weil es keinen festen Anteil an Vorlesung/Übung/Theorie/Praxis in der Klausur gibt.



# Allgemeine Fragen

## Frage:

Früher gab es viele Aufgaben, die einen größeren Rechenaufwand beinhalteten und ohne Taschenrechner schon relativ schwer waren. Ist das weiterhin so?

## Antwort:

Größere Rechenaufgaben sind möglich.

Wie immer gilt, je größer die Aufgabe, desto mehr Punkte gibt es dafür.

Es lässt sich alles ohne Taschenrechner lösen.

Lange Kommazahlen sind also unwahrscheinlich.

Bruchrechnung ist aber möglich!



# Allgemeine Fragen

## Frage:

Früher wurden bestimmte Normen vorausgesetzt (z.B. Manchester-Codierung, IEEE 802.5(Token-Ring)). Ohne zu wissen, was genau dieser Standard aussagt, konnte man die Aufgabe nicht lösen. Ist das weiterhin so?

## Antwort:

Ja, die Funktionsweise von z.B. der Manchester-Codierung sollte bekannt sein.

Es wird aber nicht nach der IEEE-Nummer vom Token-Ring gefragt.



# Allgemeine Fragen

## Frage:

Sind die Binärpräfixe überhaupt wichtig?

## Antwort:

Ja.

Dezimalpräfixe sind für uns Menschen einfacher, aber in der Informatik sind Binärpräfixe üblich.



# Allgemeine Fragen

## Frage:

Was  $T_x$  und  $T_p$  ist, ist mir klar, aber ich habe auch mal  $T_c$  gelesen.  
Wüsstest du was das sein kann?

## Antwort:

Ohne Kontext schwer zu sagen.

c könnte für collision stehen.

Oder irgendwas mit Lichtgeschwindigkeit (Formelzeichen c).



# Allgemeine Fragen

## Frage:

Müssen wir irgendwas zum Application Layer wissen? Der wurde weder in der Vorlesung noch in der Übung behandelt. Also z.B. wie HTTP funktioniert oder DNS?

## Antwort:

Den Application Layer sollte man kennen, aber es werden dazu keine Details abgefragt.

Details zu HTTP und DNS sind nicht relevant.



# Klausur: 12. August 2024, 11:00 Uhr

## Eckdaten

- 90 Minuten, 90 Punkte
- Theorie (Vorlesung, teilweise Übung)
- Praktische Anwendung (Übung)
- Keine Hilfsmittel (Taschenrechner, etc.)

## Aufgabentypen

- Wissensfragen
- Algorithmen
  - Sliding Window, Djikstra, ...
- Berechnungen
  - Bandbreite, Verzögerung, Nyquist, ...



# Klausur: 12. August 2024, 11:00 Uhr

## Räume

- Aufteilung nach Nachnamen
- Bitte seien Sie rechtzeitig da

## Sonstiges

- Denken Sie an amtlichen Lichtbildausweis und Studierendenausweis
- Sie benötigen **kein** eigenes Schreibpapier
- Lineal oder Geodreieck sind sinnvoll für Zeichnungen / Skizzen
- Antworten auf Deutsch oder Englisch
- Bei Krankheit: Ärztliches Attest



# Klausur:

# 12. August 2024

