1. Klausur SS 2015

22.07.2015

Name:	**Henze**	Vorname:	**Martin**
Studiengang:	**Das gute alte Diplom™**		
Matr.Nr.:	**267071**	Klausurnr.:	**1337**

Hinweise: (Bitte sorgfältig durchlesen!)

- Die Klausur besteht aus **7 Aufgaben auf 20 Seiten**, plus **2** zusätzliche Seiten für Notizen.
- Tragen Sie Ihre Lösungen in die dafür vorgesehenen Felder auf den Aufgabenblättern ein. Reicht der Platz nicht aus, ist für jede Aufgabe ein neues Blatt zu verwenden. Dazu können die zusätzlichen Seiten am Ende des Klausurexemplars verwendet werden. Bei Bedarf wird weiteres Papier von der Klausuraufsicht gestellt. Schreiben Sie Name, Matrikelnummer und Klausurnummer auf zusätzlich ausgehändigte Blätter und machen Sie klar, zu welcher Aufgabe eine Lösung gehört.
- Die Bearbeitungszeit beträgt **120 Minuten**.
- Die Klausur umfasst **100** Punkte. Zum Bestehen genügen **50** Punkte. Ein in den Übungen erreichter Notenbonus wird nur dann angewendet, wenn in der Klausur selbst mindestens **50** Punkte erreicht wurden.
- Am Ende der Klausur sind die Klausurblätter und evtl. zusätzlich ausgehändigte Blätter abzugeben.
- **Merken Sie Sich Ihre Klausurnummer**. Die Klausurergebnisse werden unter dieser Nummer veröffentlicht. Sie können die untere linke Ecke vorsichtig abtrennen, um die Klausurnummer mitzunehmen.
- Es sind **keine Hilfsmittel** erlaubt. Mobiltelefone sind auszuschalten. Smartwatches sind für die Dauer der Klausur bei der Aufsicht abzugeben.
- Bitte verwenden Sie keinen roten oder grünen Stift.
- Legen Sie Ihren Studierendenausweis bereit.

Ich habe die oben genannten Hinweise zur Kenntnis genommen. Ferner bestätige ich mit meiner Unterschrift, dass ich mich gesund genug fühle, an der Klausur teilzunehmen und dass ich die Aufgaben selbstständig bearbeitet habe.

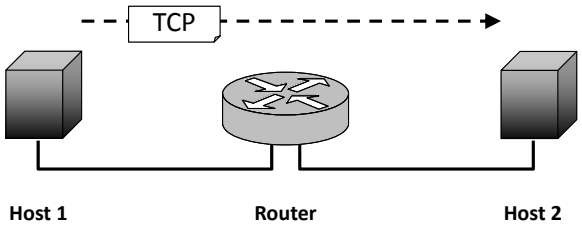
Unterschrift

Punktespiegel:

Aufgabe	1	2	3	4	5	6	7	Σ	Bonus	Note
Punkte	10	12	22	10	10	22	14	100	Noten-stufen	
davon erreicht									0.0	

Lösung 1 (Allgemeine Grundlagen) **(4 + 3 + 3) = 10 Punkte**

a) (4 Punkte) Gegeben sei ein kleines Netzwerk mit zwei Hosts, die über einen Router verbunden sind. Angenommen, $Host1$ sende ein TCP-Segment an $Host2$.

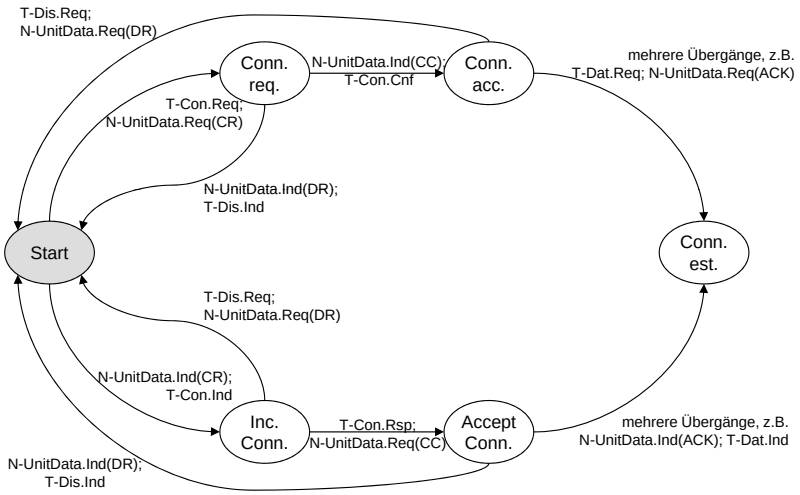


Geben Sie in *richtiger Reihenfolge* an, welche *Schichten nach dem ISO/OSI-Referenzmodell* das TCP-Segment an den jeweiligen Geräten (Hosts und Router) im Netzwerk durchläuft.

Anmerkung: es werden nicht notwendigerweise alle Zeilen benötigt.

Gerät	Schichtnummer und -name
Host 1	4. Transportschicht
Host 1	3. Vermittlungsschicht
Host 1	2. Sicherungssschicht
Host 1	1. Bitübertragungsschicht
Router	1. Bitübertragungsschicht
Router	2. Sicherungssschicht
Router	3. Vermittlungsschicht
Router	2. Sicherungssschicht
Router	1. Bitübertragungsschicht
Host 2	1. Bitübertragungsschicht
Host 2	2. Sicherungssschicht
Host 2	3. Vermittlungsschicht
Host 2	4. Transportschicht
	4 Punkte insgesamt. 0.5 Punkte Abzug für fehlende/falsche Schichten/Reihenfolge.

b) (3 Punkte) Betrachten Sie im Folgenden einen endlichen Automaten, der einen Ausschnitt des Verhaltens eines verbindungsorientierten Transportschichtprotokolls beschreibt:



i) *Welche Phase* der Verbindung beschreibt der Automat? (1 Punkt)

Phase: Verbindungsaufbau (**1 Punkt**)

ii) Das Protokoll nutzt einen *Dienst der Vermittlungsschicht*. Nennen Sie *zwei Eigenschaften*, die diesen Dienst *möglichst genau beschreiben.* (2 Punkte)

Dienst: unbestätigt (**1 Punkt**), verbindungslos (**1 Punkt**)

c) (3 Punkte) Eine Anwendung A kommuniziere über *TCP-Sockets* mit einer anderen Anwendung B.

i) Wer ist in diesem Beispiel *Dienstnehmer und wer Diensterbringer?* (1 Punkt)

Dienstnehmer:

A und B. (**0.5 Punkte**)

Diensterbringer:

Transportschicht. (**0.5 Punkte**)

ii) Was sind hier der *Dienst, das Protokoll und der Dienstzugangspunkt*? (1,5 Punkte)

Dienst:

die TCP-Funktionalität, d.h. die zuverlässige Datenübertragung. (**0.5 Punkte**)

Protokoll:

TCP (**0.5 Punkte**)

Dienstzugangspunkt:

Socket (**0.5 Punkte**)

iii) Handelt es sich bei der Kommunikation zwischen A und B um *vertikale* oder um *horizontale* Kommunikation? (0,5 Punkte)

(virtuell) horizontal (**0.5 Punkte)**.

Lösung 2 (Signale) **(1.5 + 1.5 + 0.5 + 1.5 + 2 + 5) = 12 Punkte**

a) (1.5 Punkte) *Erklären Sie knapp* den Begriff *Latenz* und nennen Sie *zwei wesentliche Faktoren*, die eine hohe Latenz bewirken.

Latenz:

Latenz: Dauer/Verzögerung zwischen Aufsetzen eines Signals vom Sender auf das Medium bis zum Eintreffen des Signals beim Empfänger. Oder einfach: Laufzeit eines Signals über ein Medium / einen Kanal.
0,5 Punkte

Faktor 1:

niedrige Ausbreitungsgeschwindigkeit bzw. Signalgeschwindigkeit
0,5 Punkte

Faktor 2:

langes Medium bzw. große Distanzen
0,5 Punkte

b) (1.5 Punkte) Welche *drei grundlegenden Modulationsarten* gibt es im *Breitbandverfahren*?

Amplituden-, Frequenz- und Phasenmodulation
0,5 Punkte pro korrekter Modulationsart. 0,5 Punkte Abzug für falsche Angaben

c) (0.5 Punkte) Welchen *Nachteil* haben Leitungscodes mit 100% Effizienz?

Mögliche Antworten: nicht selbsttaktend, nicht gleichstromfrei
0,5 Punkte

d) (1.5 Punkte) Angenommen, es steht eine *Schrittgeschwindigkeit von 200 kBaud* zur Verfügung. Geben Sie an, welche *Datenrate* mit den folgenden Leitungscodes jeweils erreicht werden kann:

i) Manchester-Code

100 kBit/s

0,5 Punkte

ii) NRZ-L-Code

200 kBit/s

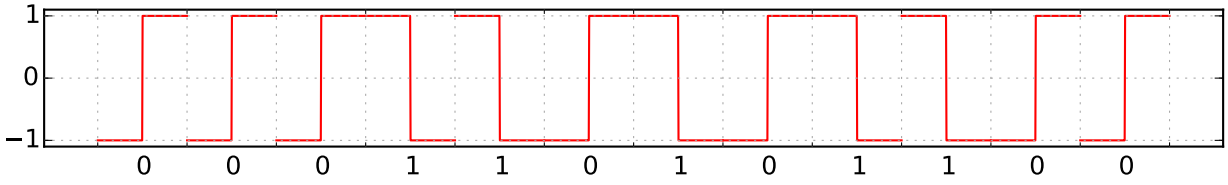
0,5 Punkte

iii) 4B/5B-Code

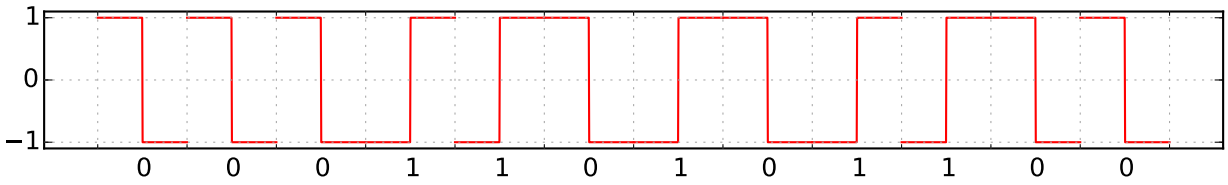
160 kBit/s

0,5 Punkte. Anmerkung: wer in den drei Teilen eine falsche Einheit verwendet, bekommt pauschal 0,5 Punkte Abzug.

e) (2 Punkte) Stellen Sie die Bitfolge **0 0 0 1 1 0 1 0 1 1 0 0** im *Manchester-Code* dar.



oder



Da der Manchester-Code nicht eindeutig ist, gibt es zwei mögliche Lösungen.
2 Punkte insgesamt; für wenige Fehler 0,5 abziehen, für mehrere 1, für noch mehr 1,5, und wenn es zu viel wird, gibt es nix mehr.

f) (5 Punkte) Gegeben sei ein Kanal mit einer Bandbreite von 5.000 Hz und einem Signal-Rauschabstand von 30 dB. Sie wollen nun 64-QAM zur Codierung Ihrer Daten auf diesem Kanal verwenden. *Ist dies unter den gegebenen Voraussetzungen möglich*? Begründen Sie Ihre Antwort mit Hilfe der *Theoreme von Shannon und Nyquist*.

Maximal erreichbare Datenraten nach Theorem:

Nyquist-Theorem: $C = B \cdot 2 \cdot ld(n)$

Shannon-Theorem: $C = B \cdot ld(1 + S/N)$

mit B: Bandbreite, n: Signalstufen und $SNR = 10 \cdot log_{10}(S/N)$

Hier: $B = 5000Hz$ und $n = 64$

$SNR = 30dB = 10 \cdot log_{10}(S/N) <=> 3 = log_{10}(S/N) <=> S/N = 10^3$
(**0,5 Punkte für Formel, 0,5 Punkte für korrekte Rechnung**)

Somit:

Nyquist: $C = 5.000 \cdot 2 \cdot ld(64) = 60.000Bit/s$
(**0,5 Punkte für Formel, 0,5 Punkte für richtige Zahl, 0,5 Punkte für richtige Einheit**)

Shannon: $C = 5.000 \cdot ld(1 + 1000) < 50.000Bit/s$
(**0,5 Punkte für Formel, 0,5 Punkte für richtige Zahl, 0,5 Punkte für richtige Einheit Wer hier = statt < sagt, bekommt die Punkte, es ist ja beinahe der Wert.**)

Also: nicht möglich, da Shannon einen kleineren Wert ergibt und die Datenrate und damit auch die Zahl der möglichen Signalstufen beschränkt. (**1 Punkt für Schlussfolgerung**)

Lösung 3 (Netzwerkschicht) **(4.5 + 13 + 1 + 0.5 + 3) = 22 Punkte**

a) (4.5 Punkte) Ein Router empfange ein IPv4-Paket mit einer Gesamtlänge von 1500 Byte und muss es auf einer Leitung mit einer MTU von 560 Byte weiterleiten. Es werden keine IP-Optionen verwendet. *Führen Sie die notwendige Fragmentierung durch*. Tragen Sie in der ersten Zeile der folgenden Tabelle die für die Fragmentierung relevanten *Headerinformationen* ein. Geben Sie in den nachfolgenden Zeilen die *zugehörigen Werte der einzelnen Fragmente* an. Es werden nicht notwendigerweise alle Zeilen und/oder Spalten benötigt.

Fragment	Payload Length	Total Length	ID	MF	Offset
1	**536 Byte**	**556**	x	**1**	**0**
2	**536 Byte**	**556**	x	**1**	**67**
3	**408 Byte**	**428**	x	**0**	**134**

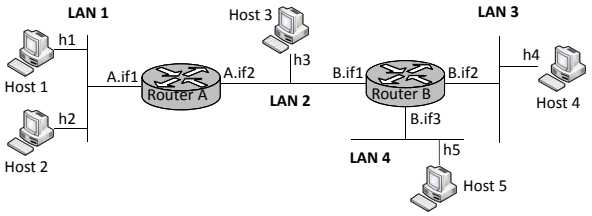
Platz für Nebenrechnungen:

Gesamtlänge heißt: 20 Byte Header + 1480 Byte Daten.
Bei den Fragmenten: Gesamtlänge von 560 Byte – 20 Byte Header + 540 Byte Daten. Bei Fragmentierung kann der Payload aber immer nur ein Vielfaches von 8 sein, also ist der maximale Payload 536 Byte. Damit ergibt sich obige Tabelle.

Bewertung (insgesamt 4,5 Punkte):

- 0,5 Punkte: Korrekte Angabe der relevanten Headerinformationen (Total Length überflüssig, ergibt sich aus Rest).
- 0,5 Punkte: Erkannt, dass IP-Header-Länge abgezogen werden muss.
- 0,5 Punkte: Berücksichtigung des Headers in jedem Fragment.
- 1,5 Punkte: Korrekte Offset-Handhabung. Wenn Offset kein Vielfaches von 8, dann nur 1 Punkt. Wer Sequenznummern verwendet bekommt noch 0,5 Punkte.
- 0,5 Punkte: Korrekte Handhabung MF.
- 0,5 Punkte: Angabe der Payloadlänge (Passend zur Offsethandhabung). Gleichmäßige Verteilung des Payloads auf drei Fragmente: 0 Punkte
- 0,5 Punkte: Verwendung der gleichen ID in allen Fragmenten.

b) (13 Punkte) Gegeben sei das in der folgenden Abbildung dargestellte Netzwerk. In jedem Subnetz (LAN 1 – LAN 4) sind exemplarisch einige Hosts dargestellt. Die Netzwerkschnittstellen (Interfaces) der Hosts sind mit `h1, ..., h5` benannt, die Netzwerkschnittstellen der Router mit der Bezeichnung des Routers und dem Zusatz `if1, if2, if3`.



i) Sie haben zur Konfiguration dieses Netzes den Adressbereich `134.130.56.0/21` erhalten. *Weisen Sie in der folgenden Reihenfolge jedem Subnetz einen IP-Adressbereich zu, indem Sie Netzadresse (Netz-ID) und Subnetzmaske angeben*:

- LAN 1: muss 1000 Rechner adressieren können
- LAN 2: muss 500 Rechner adressieren können
- LAN 3: muss 180 Rechner adressieren können
- LAN 4: muss 80 Rechner adressieren können

Wählen Sie die Subnetze jeweils so klein wie möglich und weisen Sie jeweils die niedrigstmögliche Netzadresse zu. (4 Punkte)

	Netz-ID	Subnetzmaske
LAN 1	**134.130.56.0**	**/22**
LAN 2	**134.130.60.0**	**/23**
LAN 3	**134.130.62.0**	**/24**
LAN 4	**134.130.63.0**	**/25**

Je Zeile 1 Punkt.

ii) *Teilen Sie jedem Interface der Router und der Hosts eine gültige IP-Adresse zu*. Wählen Sie dabei für die Router so niedrige Adressen wie möglich und für die Hosts so hohe Adressen wie möglich. (3 Punkte)

Interface	IP-Adresse	Interface	IP-Adresse
A.if1	**134.130.56.1**	h1	**134.130.59.254**
A.if2	**134.130.60.1**	h2	**134.130.59.253**
B.if1	**134.130.60.2**	h3	**134.130.61.254**
B.if2	**134.130.62.1**	h4	**134.130.62.254**
B.if3	**134.130.63.1**	h5	**134.130.63.126**

Je 1,5 Punkte für die Router und 1,5 Punkte für die Hosts. Je Fehler 0,5 Abzug.

iii) Zum netzübergreifenden Datenaustausch ist es erforderlich, dass die Router wissen, wie sie Pakete weiterzuleiten haben. Geben Sie dazu die *Routing-Tabellen für beide Router* an. Direkte Verbindungen können Sie wie in der Vorlesung/Übung mit einem * im Feld 'Gateway' kennzeichnen. (4 Punkte)

Router A

Ziel	Interface	Gateway
134.130.56.0/22	**A.if1**	*
134.130.60.0/23	**A.if2**	*
134.130.62.0/24	**A.if2**	**134.130.60.2**
134.130.63.0/25	**A.if2**	**134.130.60.2**

Router B

Ziel	Interface	Gateway
134.130.56.0/22	**B.if1**	**134.130.60.1**
134.130.60.0/23	**B.if1**	*
134.130.62.0/24	**B.if2**	*
134.130.63.0/25	**B.if3**	*

Je Zeile 0,5 Punkte.

iv) Host 2 verschickt ein IP-Paket an Host 5. Welchen Weg nimmt das Paket? *Geben Sie für jedes Teilstück der Strecke die Ziel-MAC-Adresse und Ziel-IP-Adresse im übertragenen Rahmen an.* (Hinweis: als Ziel-MAC-Adresse können Sie die Interface-Bezeichnungen verwenden. Sollten Sie Aufgabenteil ii) nicht gelöst haben, so tragen Sie in die dortige Tabelle beliebige IP-Adressen ein und nutzen diese hier.) (2 Punkte)

Host2 => Router A: A.if1, IP 134.130.63.126
Router A => Router B: B.if1, IP 134.130.63.126
Router B => Host 5: h5, IP 134.130.63.126
(**1 Punkt für wechselnde MAC-Adressen, dabei je 0,5 Punkte Abzug pro Fehler.**
1 Punkt für konstante IP-Adresse; 0,5 Abzug, falls es die falsche IP-Adresse ist.)

c) (1 Punkt) Welche der folgenden IPv4-Subnetzmasken ist *gültig* und welche *ungültig*? Warum?

i) `255.255.96.0`
ii) `255.255.128.0`

Die erste ist ungültig, die zweite gültig (**0.5 Punkte**). Sobald in der Subnetzmaske die erste `0` kommt, darf keine `1` mehr folgen, und dies ist nur bei der zweiten Maske gegeben (**0.5 Punkte**).

d) (0.5 Punkte) In einem Router werde ein Paket gelöscht, da seine Time-to-Live abgelaufen ist. *Welches Protokoll* wird verwendet, um eine Rückmeldung über diesen Vorfall an den Sender zu geben?

ICMP

e) (3 Punkte) *Network Address (Port) Translation (NAT)* ist eine Möglichkeit, mit der Knappheit von IP-Adressen umzugehen. Ein NAT-Router habe die folgende Abbildungstabelle angelegt:

Prot.	IP-Adresse lokal	Port lokal	IP-Adresse global	Port global	IP-Adresse Ziel	Port Ziel
TCP	10.0.0.13	6397	137.226.12.7	6397	10.0.132.67	80
TCP	137.226.12.7	4938	137.226.12.7	4938	134.130.4.33	80
TCP	10.0.0.4	5549	137.226.12.7	5549	134.130.5.4	80
TCP	10.0.0.13	4938	137.226.12.7	8539	134.130.4.33	80
TCP	10.0.0.16	6397	137.226.12.7	5549	134.130.5.4	80

Diese Abbildungstabelle ist nicht korrekt – *Korrigieren Sie alle Fehler geeignet.* Tragen Sie dazu in die unten stehende Tabelle neue (korrekte) Werte in genau die Felder ein, in denen in der obigen Tabelle fehlerhafte Werte stehen.
Bitte beachten Sie: es gibt keine eindeutige Lösung, ersetzen Sie fehlerhafte Einträge (und auch nur diese) lediglich durch im Kontext passende Einträge.

Prot.	IP-Adresse lokal	Port lokal	IP-Adresse global	Port global	IP-Adresse Ziel	Port Ziel
					(3)	
	(1)					
				(2)		
				(2)		

Je gefundenem Fehler 0,5 Punkte, je korrekter Korrektur 0,5 Punkte:

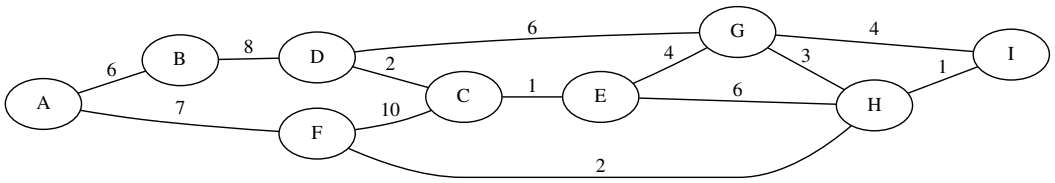
(1) Hier darf keine globale Adresse stehen. Ersetzung durch beliebige Adresse aus dem 10er-Netz, außer 10.0.0.13, da für diese Adresse der verwendete Port schon existiert.

(2) Bei „Port global“ haben zwei Einträge den gleichen Port (5549). Einer davon muss ersetzt werden durch einen beliebigen anderen, der noch nicht in Verwendung ist.

(3) Die Ziel-IP-Adresse kann keine lokale Adresse sein. Sie muss durch eine beliebige global gültige Adresse ersetzt werden.

Lösung 4 (Routing) **(6 + 4) = 10 Punkte**

a) (6 Punkte) Gegeben sei das folgende Netzwerk, in dem *Link-State-Routing* verwendet wird. Die Knoten stellen Router dar, die Kanten Leitungen zwischen den Routern und die Beschriftungen der Kanten ein Maß für die Kosten der Übertragung auf der entsprechenden Leitung.



Berechnen Sie mit Hilfe des Djikstra-Algorithmus' alle kürzesten Pfade von A nach C. Ergänzen Sie dazu die folgende Tabelle, indem Sie spaltenweise die Einzelschritte des Algorithmus' dokumentieren. Verwenden Sie Einträge der Form n, X. Dabei gibt $n \in \mathbb{N}$ die Kosten des kürzesten Pfades zum betrachteten Knoten und $X \in \{A, \ldots, I\}$ den bzw. die Vorgänger an. Ein Kasten um einen Eintrag markiert den im jeweiligen Schritt bestimmten Arbeitsknoten.

Es werden nicht notwendigerweise alle Spalten der Tabelle benötigt.

	Schritt								
Router	0	1	2	3	4	5	6	7	8
A	**0,-**								
B	∞	**6,A**							
C	∞	∞	∞	17,F	17,F	17,F	17,F	16,D	**16,D;E**
D	∞	∞	14,B	14,B	14,B	14,B	**14,B**		
E	∞	∞	∞	∞	15,H	15,H	15,H	**15,H**	
F	∞	7,A	**7,A**						
G	∞	∞	∞	∞	12,H	**12,H**			
H	∞	∞	∞	**9,F**					
I	∞	∞	∞	∞	**10,H**				

Geben Sie die berechneten kürzesten Pfade von A nach C sowie deren Kosten an:

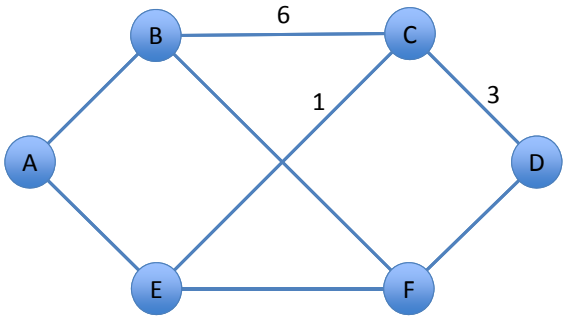
Es gibt zwei Pfade, die Kosten sind für alle 16:

1) A, B, D, C

2) A, F, H, E, C

Bewertung: 5 Punkte für Tabelle; jeweils 0,5 Abzug für Rechenfehler, je 1 Abzug für Wahl des falschen Knotens, Erhöhung von Werten oder Hinzufügen nicht existierender Einträge.
1 Punkt für Auflistung aller kürzesten Pfade konsistent zur Tabelle.

b) (4 Punkte) Gegeben sei das folgende Netzwerk, in dem *Distance-Vector-Routing* verwendet wird:



Router C habe gerade die folgenden Abstandsvektoren empfangen:

- von Router B: $DV_B =$ `((A,1), (B,0), (C,6), (D,9), (E,3), (F,3))`
- von Router D: $DV_D =$ `((A,7), (B,9), (C,3), (D,0), (E,7), (F,1))`
- von Router E: $DV_E =$ `((A,2), (B,3), (C,1), (D,7), (E,0), (F,4))`

Diese Abstandsvektoren geben jeweils die Pfadkosten der Quelle des jeweiligen Vektors hin zu allen anderen Knoten im Netz (A bis F) an.

Als Kostenmaß wird die aktuelle Auslastung der Knoten verwendet. Die zugehörige Metrik ist additiv, d.h. zur Berechnung der Güte eines Pfades werden die Einzelwerte aufaddiert. Pfade mit kleineren Werten werden bevorzugt. Die aktuelle Auslastung der Nachbarn von C ist im obigen Netzwerk dargestellt.

i) *Geben Sie die Routing-Tabelle an, die C aufgrund dieser Informationen berechnet.* (3 Punkte)

Ziel	Next Hop	Kosten
A	E	3
B	E	4
C	-	0
D	D	3
E	E	1
F	D	4

Platz für Nebenrechnungen:

B

Ziel	Kosten
A	1+6=7
B	0+6=6
C	6+6=12
D	9+6=15
E	3+6=9
F	3+6=9

D

Ziel	Kosten
A	7+3=10
B	9+3=12
C	3+3=6
D	0+3=3
E	7+3=10
F	1+3=4

E

Ziel	Kosten
A	2+1=3
B	3+1=4
C	1+1=2
D	7+1=8
E	0+1=1
F	4+1=5

Cs Routingtabelle

Ziel	Next Hop	Kosten
A	E	3
B	E	4
C	-	0
D	D	3
E	E	1
F	D	4

Keine Rechnung nötig; Endtabelle reicht. Pro korrekter Zeile 0,5 Punkte. Die Zeile zu C braucht nicht da zu sein – ist sie nicht angegeben, gibt es auch die 0,5 Punkte.

ii) Geben Sie an, *welchen Abstandsvektor C auf Basis seiner Routing-Tabelle versendet.* (1 Punkt)

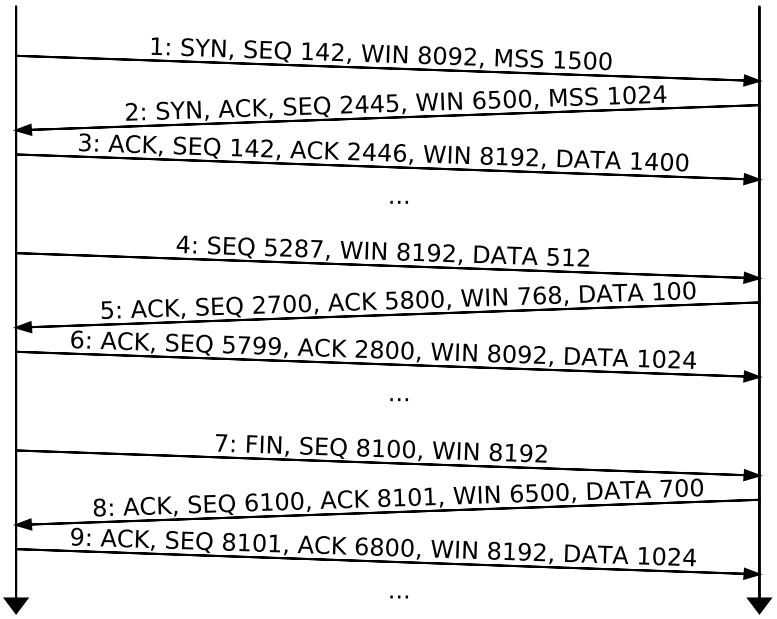
$DV_B =$ `((A,3), (B,4), (C,0), (D,3), (E,1), (F,4))`
1 Punkt. Nur 0,5 Punkte für 1 oder 2 Fehler, sonst 0 Punkte. Folgefehler beachten!

Lösung 5 (Transportschicht) **(3.5 + 4.5 + 1 + 1) = 10 Punkte**

a) (3.5 Punkte) Gegeben sind die unten stehenden (fehlerhaften) Auszüge einer TCP-Verbindung. Das Format ist dabei:

`<N>: {<FLAG>,}* SEQ <S>, [ACK <A>,] WIN <W>, [MSS <M>,] [DATA <D>]`

wobei `N` die Segmente lediglich zu Referenzzwecken durchnummeriert. Mit `FLAG` werden die Flags SYN, ACK und FIN genau dann angegeben, wenn sie gesetzt sind. `S` ist die Sequenznummer, `A` die Bestätigungsnummer genau dann wenn gesetzt, `W` die Window Size und `M` die Maximum Segment Size sofern gesetzt. Wenn `DATA <D>` angegeben ist, enthält die Nachricht `D` Byte Payload. "..." steht für beliebig viele Segmente, die hier nicht dargestellt sind. Das Auslesen von Daten aus den Empfangspuffern durch die Applikationen ist hier nicht dargestellt.



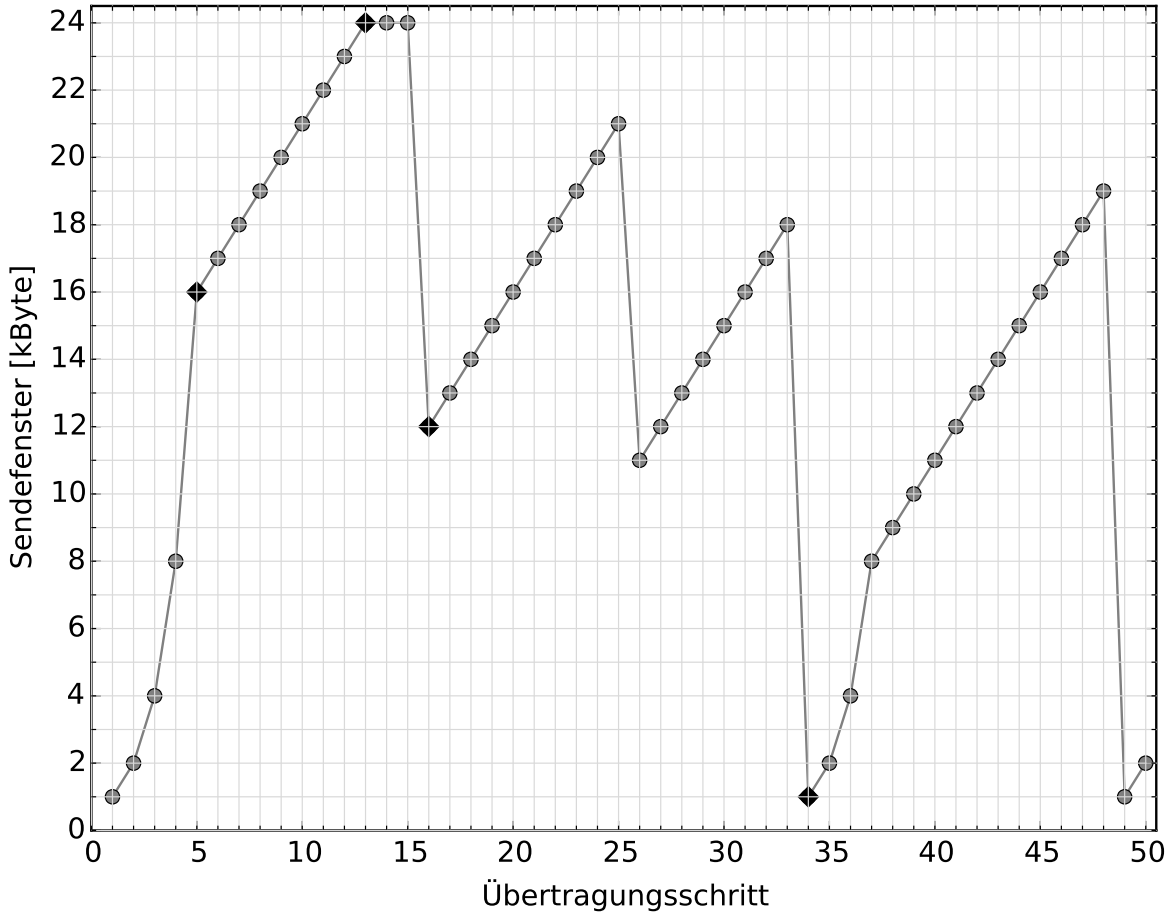
Identifizieren Sie sieben Fehler in der obenstehenden Kommunikation. Beziehen Sie sich bei der Nennung der Fehler auf die Referenznummern der Segmente.

Hinweis: Die Segmente 1, 4 und 7 werden als korrekt angenommen und brauchen nicht überprüft zu werden. In jeder anderen Nachricht können kein, ein oder mehrere Fehler enthalten sein.

- 2: Hier fehlt die ACK-Nummer 143.
- 3: Die Sequenznummer muss inkrementiert werden, also 143.
- 3: Der Empfänger gibt eine MSS von 1024 an, wir dürfen also nur 1024 Byte senden.
- 5: Das Fenster in 2 erlaubt bereits Übertragungen bis 6643, wir können es hier nicht auf 6568 beschränken.
- 6: Der Empfänger hat alles bis 5800 bestätigt, wir sollten also 5799 nicht noch einmal übertragen. Oder alternative Möglichkeit – 5: es muss ACK 5799 heißen.
- 6: Weder das Fenster aus 5 noch das Fenster aus 2 erlauben, 1024 Byte Daten zu senden.
- 9: Wir haben schon ein FIN gesetzt, wir dürfen keine Daten mehr senden.

0,5 Punkte pro richtigem Fehler. Falls mehr als 7 Fehler angegeben sind, gibt es 0,5 Punkte Abzug für falsche Fehler.

b) (4.5 Punkte) Gegeben ist das folgende Diagramm einer TCP-Datenübertragung, in der vereinfachend davon ausgegangen wird, dass die Übertragung in einzelnen Schritten stattfindet. Das Diagramm zeigt für jeden Schritt die Menge an Daten, die der Sender versenden darf.



Beantworten Sie folgende Fragen:

i) (2 Punkte) Was passiert an den markierten Stellen (Übertragunsschritt 5, 13, 16 und 34)?

ii) (0,5 Punkte) Wie groß ist der Threshold im 20. Übertragungsschritt?

iii) (0,5 Punkte) Wie groß ist der Threshold im 35. Übertragungsschritt?

iv) (0,5 Punkte) Wie groß muss der Puffer des Empfängers mindestens sein?

v) (1 Punkt) Wie groß ist die maximal erreichbare Datenrate, wenn die RTT 12 ms beträgt?

i)
- 5: Threshold erreicht.
- 13: Receiver-Window erreicht.
- 16: Triple-DUP-ACK, Fast Retransmit.
- 34: Timeout, zurück zu Slow-Start.

ii) 12 kB

iii) 9 kB (ja, ich weiß, der war gemein ;))

iv) 24 kB

v) $\frac{24\,\mathrm{kB}}{12\,\mathrm{ms}} = 2\,\mathrm{Mb/s}$ (**falls das Rechenergebnis nicht angegeben ist, nur 0,5 Punkte, sonst 1 Punkt.**)

c) (1 Punkt) *Führt ein verlorengegangenes ACK bei TCP stets zu einer Übertragungswiederholung?* Begründen Sie Ihre Antwort.

Nein. (**0,5 Punkte**)
ACKs sind kumulativ. (**0,5 Punkte**)
(Oder statt Erwähnung der Kumulation: kommt nach dem verlorengegangenen ACK noch ein ACK für ein späteres Segment an, ist das vorherige Segment direkt mitbestätigt und braucht nicht neu übertragen zu werden.)

d) (1 Punkt) Angenommen, auf allen Links im Internet würden Daten zuverlässig übertragen. *Wäre die Implementierung eines zuverlässigen Datenübertragungsdienstes durch TCP dann überflüssig*? Begründen Sie Ihre Antwort.

Nein. (**0,5 Punkte**)
Nach jedem Hop durchlaufen die Pakete einen Router. Dieser kann Daten verwerfen (Congestion, RED) bzw. Daten über unterschiedliche Links weiterleiten.
Es können also trotzdem Daten verlorengehen oder in falscher Reihenfolge ankommen. (**0,5 Punkte**)
Dafür wird etwas auf einer Schicht über IP benötigt, also bei TCP.

Lösung 6 (Sicherungsschicht) **(4 + 6 + 5 + 7) = 22 Punkte**

a) (4 Punkte) Ein Sender möchte die folgende Bitsequenz übertragen: `10100011`. Er sichert die Sequenz mit einer *Cyclic Redundancy Checksum (CRC)* mit dem Generatorpolynom

$$G(x) = x^5 + x^3 + x^2 + 1$$

Der Empfänger empfängt die folgende Bitsequenz: `1010001111100`.

Beantworten Sie folgenden Fragen und begründen Sie Ihre Antworten:

i) *Wurden die Datenbits korrekt übertragen*? (1 Punkt)

Offensichtlich ja. Denn die empfangenen Daten entsprechen ohne CRC den gesendeten.
0,5 Punkte für Ja; 0,5 Punkte für Begründung.

ii) *Wie handelt der Empfänger bei Erhalt der Bitsequenz*? (3 Punkte)

Der Empfänger führt erstmal den CRC durch:

```
|1010001111100
|101101
|   101111
|   101101
|       101100
|       101101
|            1
```

2 Punkte für die Polynomdivision. Falls diese bereits in i) gemacht wurde, auch die Punkte geben.

Der Rest ist $1 \neq 0$. Also geht der Empfänger von einer fehlerhaften Übertragung aus und verwirft das Paket. (Hier ist also wohl ein Fehler in der Prüfsumme aufgetreten. Wichtig ist, dass die Studenten erkennen, dass der Empfänger dies nicht von einem Fehler in den Daten unterscheiden kann und daher das Paket verwirft.) **1 Punkt**

b) (6 Punkte) Eine Möglichkeit zur Fehlerkorrektur ist der Einsatz des *Hamming-Codes*.

Sie haben die folgenden beiden Bitsequenzen erhalten, die mit dem Hamming-Code geschützt sind. Überprüfen Sie für beide Sequenzen, ob es zu Übertragungsfehlern gekommen ist. ***Kreisen*** *Sie dazu die Prüfbits, für die Sie andere Werte berechnen,* ***ein*** *und* ***unterstreichen*** *Sie diejenigen Bits, die Sie folglich als falsch identifizieren. Schreiben Sie die (korrigierten) Datenbits in die rechte Spalte der Tabelle.*

Hamming-codierte Daten									Datenbits
1	2	3	4	5	6	7	8	9	
(0)	(0)	0	0	1	1	0	0	0	**1 1 1 0 0**
0	(0)	0	1	0	1	0	0	0	**0 0 1 0 0**

Für jede Bitfolge 3 Punkte: 1 für falsche Prüfbits, einen für markierte Bits, einen für extrahierte Datenbits.

c) (5 Punkte) Gegeben sei ein Netzwerk mit einer Bustopologie und einer Ausdehnung von 300 m, in dem CSMA/CD eingesetzt wird. Die Datenrate betrage 900 MBit/s, die Signalgeschwindigkeit im physikalischen Medium sei $2 \cdot 10^8$ m/s.

i) Wieviel *Zeit* kann *maximal* vergehen, bis eine *sendende Station* eine *Kollision erkennt*? (2 Punkte)

Zeit im Worst-Case gleich doppelter Signallaufzeit zur Gegenseite:

$$t_{\text{max}} = \frac{2 \cdot 300\,\text{m}}{2 \cdot 10^8\,\text{m/s}} = \frac{3 \cdot 10^2}{10^8}\,\text{s} = 3 \cdot 10^{-6}\,\text{s}$$

Bewertung: 0,5 Punkte für Berücksichtigung des Faktors 2, 0,5 Punkte für korrekte sonstige Formel, 1 Punkt für richtige Rechnung.

ii) Welche *minimale Rahmenlänge* wäre für dieses LAN erforderlich? (2 Punkte)

Minimale Rahmenlänge: wieviel kann man in $3 \cdot 10^{-6}$ Sekunden versenden?

$$r_{\text{min}} = 900\,\text{MBit/s} \cdot 3 \cdot 10^{-6}\,\text{s} = 10^6 \cdot 2700 \cdot 10^{-6}\,\text{Bit} = 2700\,\text{Bit} = 337.5\,\text{Byte}$$

(oder 338 Byte, wenn man nur ganze Bytes verschicken will)
Bewertung: 1 Punkt für Multiplikation des Ergebnisses aus der vorherigen Teilaufgabe mit der Strecke, 1 Punkt für richtige Rechnung. (Folgefehler beachten!)

iii) Sie wechseln nun zu einer Sterntopologie mit *Switch* (im Full-Duplex-Mode), behalten aber Datenrate und Ausdehnung bei. *Ändert sich die erforderliche minimale Rahmenlänge?* Begründen Sie Ihre Antwort. (1 Punkt)

Bei einem Switch (im Full-Duplex-Mode) können keine Kollisionen mehr auftreten, drum ist die Festlegung einer minimalen Rahmenlänge unerheblich.
Wer hier sagt, die minimale Rahmenlänge wäre Null oder würde der Headerlänge entsprechen, darf den Punkt auch bekommen. Wer was Schwammiges schreibt, bekommt nur 0,5 Punkte. Wer Blödsinn schreibt, bekommt gar nix.

d) (7 Punkte) Ein Knoten A möchte an einen benachbarten Knoten B Daten übertragen. Die Latenz des Links betrage 1.8 ms, die maximale Datenrate des Full-Duplex-Links sei in jede Richtung 16 Mbit/s. Die Headerinformationen eines Rahmens seien 16 Byte groß, ein Acknowledgment-Rahmen (ACK) habe eine Gesamtgröße von 16 Byte. Pro Rahmen können maximal 768 Byte Nutzdaten übertragen werden.

i) Welche *Nutzdatenrate* lässt sich für die Datenübertragung von A nach B unter Verwendung von *Stop-and-Wait* in diesem Szenario *maximal* erreichen, wenn *keine Bitfehler* auftreten? Die Verarbeitungszeit auf Empfängerseite soll vernachlässigt werden, d.h. der Empfänger kann ein ACK direkt nach Erhalt eines Rahmens absenden. (3 Punkte)

Um 768 Byte Nutzdaten zu übertragen, müssen wir einen Frame senden und auf die Antwort warten. Es kommen also noch 32 Byte Kontrollinformationen dazu. Insgesamt benötigen wir also die Zeit, um 800 Byte auf den Kanal zu legen + 2 mal das Link-Delay:

$$\frac{800 \cdot 8\,\text{bit}}{16\,\text{Mbit/s}} + 3.6\,\text{ms} = 4\,\text{ms} \tag{1}$$

0,5 Punkte für Umrechnung Byte in Bit; 0,5 Punkte für Berücksichtigung Sendezeit; 0,5 Punkte für Berücksichtigung doppelte Latenz; 0,5 Punkte für korrekte Rechnung

In dieser Zeit übertragen wir also 768 Byte Nutzdaten. Also liegt die Nutzdatenrate bei:

$$\frac{768 \cdot 8\,\text{bit}}{4\,\text{ms}} \approx 1.536\,\text{Mbit/s} \tag{2}$$

0,5 Punkte für Formel; 0,5 Punkte für korrekte Rechnung

ii) Nehmen Sie nun an, die Bitfehlerrate auf dem gegebenen Kanal liege bei 10^{-4} und die Bitfehler seien nicht korreliert. Wie groß ist die *Paketfehlerrate* für einen *Rahmen maximaler Länge*? Wie groß ist die *Paketfehlerrate* für einen *Acknowledgment-Rahmen*?

Hinweis: sie können das exakte Ergebnis ohne Taschenrechner nicht berechnen. Vereinfachen Sie den Ausdruck soweit wie möglich. (1.5 Punkte)

Wenn die Fehlerrate bei 10^{-4} liegt, ist die Wahrscheinlichkeit, dass das Bit erfolgreich ankommt $1 - 10^{-4} = 0,9999$. Die Wahrscheinlichkeit, dass n Bits korrekt ankommen, liegt also bei $0,9999^n$. Für 784 Byte = 6272 Bit ergibt sich also $0.9999^{6272} \approx 0.534$ als Erfolgswahrscheinlichkeit, also eine Fehlerrate von $1 - 0.9999^{6272}$.
Für den ACK-Frame (128 bit) ergibt sich eine Fehlerrate von $1 - 0.9999^{128}$.
0,5 Punkte für allgemeine Formel, je 0,5 Punkte für Einsetzen der korrekten Werte.

iii) Zusätzlich zu den positiven Bestätigungen (ACK) gebe es nun auch negative Bestätigungen (NAK), die der Empfänger eines Rahmens sendet, wenn der Rahmen Bitfehler enthält. NAK-Rahmen haben ebenfalls eine Gesamtgröße von 16 Byte. Es gehen grundsätzlich keine Rahmen verloren, d.h. auf jeden Übertragungsversuch folgt entweder ein ACK oder ein NAK. Ein verfälschtes ACK werde ebenfalls als NAK interpretiert. Zur Vereinfachung sei angenommen, dass Bitfehler immer erkannt werden, so dass verfälschte Rahmen zuverlässig erkannt werden und ein verfälschtes NAK nie als ACK interpretiert werden kann. Folgende Paketfehlerraten seien bekannt:

Paketgröße (Byte/Bit)	Paketfehlerrate
$1(Byte) \cdot 8 = 8(Bit)$	0.1 %
$2 \cdot 8 = 16$	0.2 %
$4 \cdot 8 = 32$	0.3 %
$8 \cdot 8 = 64$	0.6 %
$16 \cdot 8 = 128$	1 %
$32 \cdot 8 = 256$	3 %
$96 \cdot 8 = 768$	7.3 %
$97 \cdot 8 = 776$	7.4 %
$98 \cdot 8 = 784$	7.5 %
$100 \cdot 8 = 800$	7.7 %
$768 \cdot 8 = 6144$	45 %
$776 \cdot 8 = 6208$	46 %
$784 \cdot 8 = 6272$	47 %
$800 \cdot 8 = 6400$	48 %

Wie groß ist die mittlere Nutzdatenrate beim Einsatz von Stop-and-Wait, wenn vom Versenden eines Rahmens bis zum Erhalt der positiven oder negativen Quittung 8 ms vergehen? (2,5 Punkte)

Wir müssen zunächst bestimmen, wie groß die Wahrscheinlichkeit ist, dass eine Übertragung schief geht. Dazu genügt es, wenn ein Paket *oder* ACK verfälscht werden. Wir berechnen also die Wahrscheinlichkeit, dass das Ereignis *Paket verfälscht* oder das Ereignis *ACK verfälscht* eintritt:

$$p(\text{Fehler}) = 1 - (1 - 0.47) \cdot (1 - 0.01) \tag{3}$$

$$p(\text{Fehler}) = 1 - 0.53 \cdot 0.99 = 1 - 0.5247 = 0.4753 \approx 0.48 \tag{4}$$

Alternativ: Damit eine Übertragung funktioniert, müssen insgesamt $800 \cdot 8\,\text{bit}$ korrekt übertragen werden. Also ergibt sich durch Ablesen aus der Tabelle:

$$p(\text{Fehler}) = 0.48 \tag{5}$$

Bis hierher 1 Punkt, entweder durch Rechnung oder durch korrektes Ablesen.
Im Mittel sind also 52 % aller Übertragungen erfolgreich. Pro Sendeversuch schaffen wir also nicht mehr 768 Byte Nutzdaten, sondern im Mittel $768 \cdot 0.52\,\text{Byte} \approx 399\,\text{Byte}$.
0,5 Punkte für Erkennen, dass man 52% berechnen muss
Also ergibt sich als mittlere Nutzdatenrate:

$$\frac{399 \cdot 8\,\text{bit}}{8\,\text{ms}} \approx 399\,\text{Mbit/s} \tag{6}$$

0,5 Punkte für Formel, 0,5 Punkte für Rechnung.

Lösung 7 (Sicherheit) **(5 + 5 + 4) = 14 Punkte**

a) (5 Punkte) Berechnen Sie einen geheimen Schlüssel zwischen Alice und Bob unter Verwendung des Algorithmus' von Diffie-Hellman. Es seien $p = 7$ und $g = 5$ gegeben. Alice verwendet den Geheimwert $a = 2$, Bob den Geheimwert $b = 3$.

i) *Geben Sie an, welche Operationen Alice und Bob jeweils ausführen und welche Informationen an den jeweiligen Kommunikationspartner übermittelt werden.* (3 Punkte)

1.) Alice wählt $a = 2$ und berechnet $A = g^a \bmod p = 5^2 \bmod 7 = 4$. **0,5 Punkte.**
2.) Alice sendet A an Bob. **0,5 Punkte.**
3.) Bob wählt $b = 3$, berechnet $B = g^b \bmod p = 5^3 \bmod 7 = 6$ **0,5 Punkte.**
4.) Bob sendet B an Alice. **0,5 Punkte.**
5.) Bob berechnet den gemeinsamen Schlüssel $K = A^b \bmod p = 4^3 \bmod 7 = \mathbf{1}$. **0,5 Punkte**.
6.) Alice berechnet den Schlüssel $K = B^a \bmod p = 6^2 \bmod 7 = \mathbf{1}$. **0,5 Punkte**.

Achtung! die Schritte brauchen nicht notwendigerweise in dieser Reihenfolge zu erfolgen! Lösung auch mit Zeichnung möglich.

ii) *Begründen Sie, warum beim Schlüsselaustausch Probleme auftreten können und skizzieren Sie einen entsprechenden Angriff.* (2 Punkte)

Die Nachrichten sind nicht authentifiziert (**0,5 Punkte**). Dadurch ist ein Man-in-the-Middle-Angriff möglich – wer den hier skizziert, bekommt die anderen **1,5 Punkte**.

b) (5 Punkte) Gegeben sind die Primzahlen $p = 13$ und $q = 7$. Zur RSA-Verschlüsselung wird der Wert $e = 5$ gewählt.

Berechnen Sie den öffentlichen Schlüssel $\langle e, n \rangle$ *und den privaten Schlüssel* $\langle d, n \rangle$. Verwenden Sie für die Berechnung von d z.B. den erweiterten euklidischen Algorithmus.

$$n = p \cdot q = 13 \cdot 7 = 91$$

$$\Rightarrow \langle e, n \rangle = \langle 5, 91 \rangle$$

0,5 Punkte für Formel, 0,5 Punkte für Wert

$$\Phi(n) = (p-1) \cdot (q-1) = 12 \cdot 6 = 72$$

0,5 Punkte für Formel, 0,5 Punkte für Wert
Bestimmung des Inversen: durch erweiterten euklidischen Algorithmus. Oder was anderes. Oder durch scharfes Hinsehen, das geht hier auch gut. Wie auch immer: das multiplikative Inverse ist 29.

$$\Rightarrow \langle d, n \rangle = \langle 29, 91 \rangle$$

Insgesamt 3 Punkte für korrektes Ergebnis. Rechenweg egal.

c) (4 Punkte)

(i) Alice besitzt ein Dokument, das sie jedem sendet, der darum bittet. Hunderte von Personen wollen dieses Dokument erhalten, doch jeder möchte auch sicher sein, dass das erhaltene Dokument tatsächlich von Alice kommt. *Welches Verfahren sollte Alice in diesem Fall verwenden, um die Authentizität des Dokuments sicherzustellen*: ein Verfahren, das auf *digitalen Signaturen* basiert (wie z.B. RSA oder DSA) oder ein Verfahren, das einen *Message Authentication Code* verwendet (der z.B. durch AES/CBC berechnet werden kann)? Begründen Sie Ihre Antwort. (1 Punkt)

Digitale Signaturen. (**0,5 Punkte**)
Geringerer Aufwand ; keine Schlüsselverteilung nötig ; ... (nur ein Grund nötig, **0,5 Punkte**)

(ii) *Warum wird bei der Berechnung digitaler Signaturen üblicherweise nur der Hash-Wert einer Nachricht signiert, nicht die Nachricht selbst?* (1 Punkt)

Asymmetrische Verfahren sind sehr rechenaufwändig. Das Berechnen und Signieren eines Hash geht deutlich schneller. (**1 Punkt, Abzüge bei schwammiger Formulierung**)

(iii) Die Firma SecureSys bietet ihre Sicherheitssoftware S auf ihrer Webseite zum Download an. Der Webserver hat ein Zertifikat mit einem öffentlichen Schlüssel; der zugehörige private Schlüssel ist sicher hinterlegt. Der Systemadministrator schlägt folgende Authentifizierungsmethode für den vertrauenswürdigen Download der Software vor: Zusammen mit Software S wird der folgende Wert im Download zur Verfügung gestellt: $H(S||KU_W)$.

Es gilt die folgende Syntax:

- S: Software
- KR_W: privater Schlüssel des SecureSys-Webservers
- KU_W: öffentlicher Schlüssel des SecureSys-Webservers
- $H(M)$: kryptographischer Hash von M
- $||$: Konkatenation zweier Bitfolgen

Ist die vorgeschlagene Methode geeignet, um den Download zu authentifizieren? Wenn ja, warum? Wenn nein, warum nicht und was sollte stattdessen getan werden? (2 Punkte)

Die Methode ist nicht geeignet, denn jeder der die Hashfunktion und den öffentlichen Schlüssel kennt, kann die Authentifizierungsinformationen fälschen.
0,5 Punkte für nein. 0,5 Punkte für Grund.

Der Hashwert sollte mit dem privaten Schlüssel verschlüsselt werden! Wobei da schon der Hashwert über S reicht. Oder alternativ kann man auch die ganze Software signieren (auch wenn es nicht empfehlenswert ist, siehe vorherige Teilaufgabe).
1 Punkt für korrekte Lösung. 0,5 Abzug, wenn schwammig formuliert.