

Name

Vorname

Studiengang (Hauptfach)

Fachrichtung (Nebenfach)

Matrikelnummer

Unterschrift der Kandidatin/des Kandidaten

.....
Note

TECHNISCHE UNIVERSITÄT MÜNCHEN

Fakultät für Informatik

- Midterm-Klausur
- Final-Klausur

- Semestralklausur
- Diplom-Vorprüfung
- Bachelor-Prüfung
-

- Einwilligung zur Notenbekanntgabe per E-Mail / Internet

Prüfungsfach: Grundlagen Rechnernetze und Verteilte Systeme

Prüfer: Prof. Dr.-Ing. Georg Carle

Datum: 09.08.2013

Hörsaal:

Reihe:

Platz:

	I	II
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		

Σ		
---	--	--

Nur von der Aufsicht auszufüllen:

Hörsaal verlassen von : bis :

Vorzeitig abgegeben um :

Besondere Bemerkungen:



Endterm-Klausur

Grundlagen Rechnernetze und Verteilte Systeme

Prof. Dr.-Ing. Georg Carle
Lehrstuhl für Netzarchitekturen und Netzdienste
Fakultät für Informatik
Technische Universität München

Freitag, 09.08.2013
11:30 – 13:00 Uhr

- Diese Klausur umfasst **23 Seiten** und insgesamt **6 Aufgaben**. Bitte kontrollieren Sie jetzt, dass Sie eine vollständige Angabe erhalten haben.
- Schreiben Sie bitte in die Kopfzeile **jeder Seite** Namen und Matrikelnummer.
- Schreiben Sie weder mit roter / grüner Farbe noch mit Bleistift.
- Die Gesamtzahl der Punkte beträgt 85.
- Als Hilfsmittel sind **ein beidseitig handschriftlich beschriebenes DIN A4-Blatt** sowie **ein nicht-programmierbarer Taschenrechner** zugelassen. Bitte entfernen Sie alle anderen Unterlagen von Ihrem Tisch und schalten Sie Ihre Mobiltelefone aus.
- Mit * gekennzeichnete Aufgaben sind ohne Kenntnis der Ergebnisse vorhergehender Teilaufgaben lösbar.
- Halten Sie sich bei der Bearbeitung nicht zulange mit einer (Teil-)Aufgabe auf. Wenn Sie die Aufgabe nicht sofort lösen können, machen Sie lieber mit der nächsten Aufgabe weiter.
- **Es werden nur solche Ergebnisse gewertet, bei denen ein Lösungsweg erkennbar ist.** Textaufgaben sind **grundsätzlich zu begründen**, falls es in der jeweiligen Teilaufgabe nicht ausdrücklich anders vermerkt ist.

Aufgabe 1 Abtastung und Quantisierung (18 Punkte)

Gegeben sei das zeit- und wertkontinuierliche Basisbandsignal auf Empfängerseite aus Abbildung 1.1. Dieses Signal soll im Verlauf dieser Aufgabe abgetastet, quantisiert und schließlich dekodiert werden.

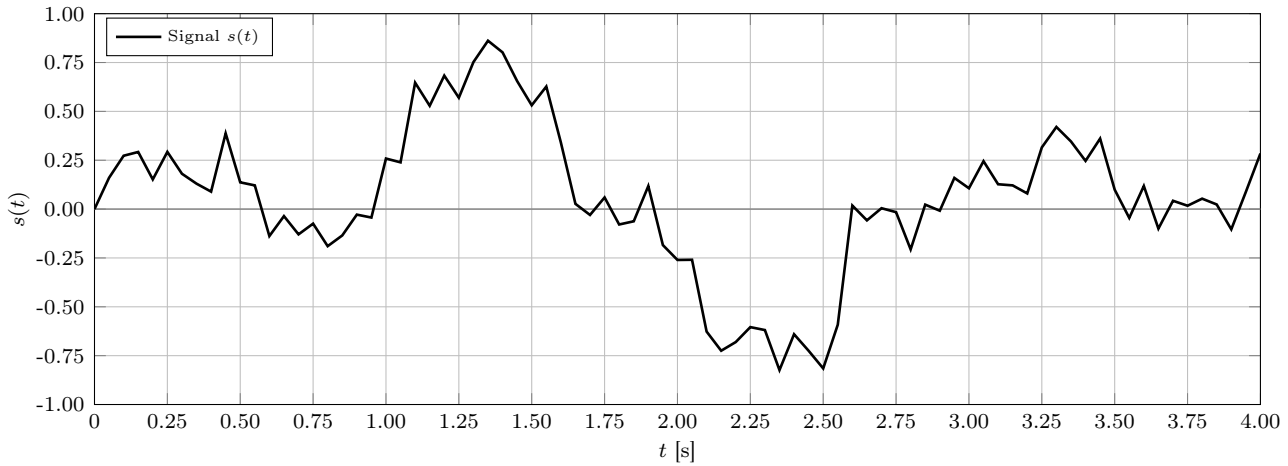


Abbildung 1.1: Zeit- und wertkontinuierliches Signal auf Empfängerseite

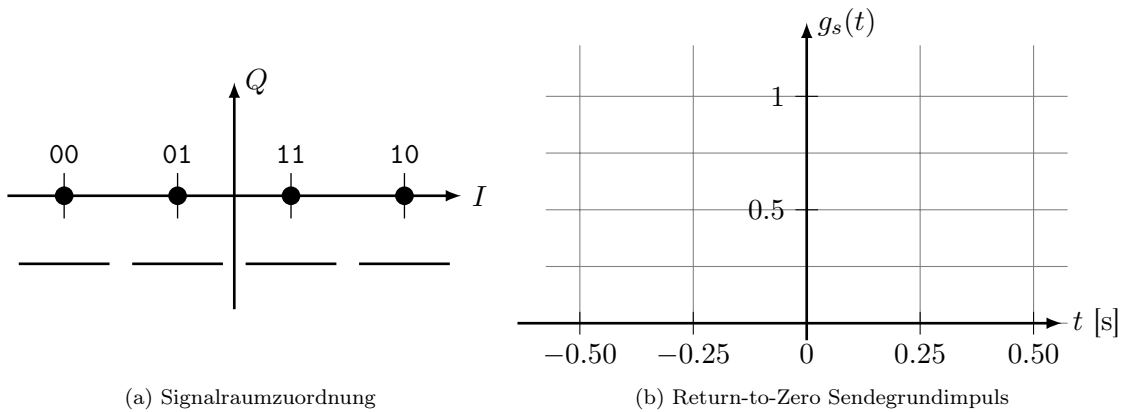
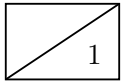


Abbildung 1.2: Sendegrundimpuls und Signalraumzuordnung

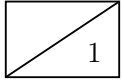
Es sei bekannt, dass es sich in Abbildung 1.1 um ein RZ-kodiertes (Return-to-Zero) Signal handelt. Das verwendete Modulationsalphabet ist in Abbildung 1.2a dargestellt.

a)* Nennen Sie zwei Kanaleinflüsse, die im Signalverlauf von Abbildung 1.1 sichtbar sind.

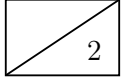
b)* Um welches Modulationsverfahren handelt es sich gemäß Abbildung 1.2a?



c)* Zeichnen Sie in Abbildung 1.2b den Return-to-Zero Sendegrundimpuls mit auf 1 normierter Amplitude im Zeitbereich $t \in [-0.5, 0.5]$ ein.



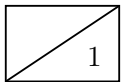
d) Angenommen der Grundimpuls aus Abbildung 1.2b soll mit der Abtastfrequenz $f_a = 2$ Hz abgetastet werden. Markieren Sie die Abtastzeitpunkte direkt in Abbildung 1.2b.



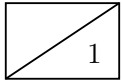
e) Tasten Sie das Signal aus Abbildung 1.1 mit der Abtastfrequenz $f_a = 2$ Hz im Bereich $t \in [0, 4]$ ab. Tragen Sie das zeitdiskrete Signal direkt in Abbildung 1.1 ein.

Hinweis: Achten Sie auf korrekte Abtastzeitpunkte gemäß Ihrer Lösung von Teilaufgabe c). Die Nutzung eines Geodreicks ist im Übrigen erlaubt.

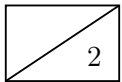
Das Quantisierungsintervall, innerhalb dessen der Quantisierungsfehler minimiert werden soll, sei $I = [-1, 1]$. Die Schrittweite betrage $\Delta = 0.5$. Zur Quantisierung werde arithmetisches Runden verwendet.



f)* Bestimmen Sie rechnerisch die Anzahl der Quantisierungsstufen.

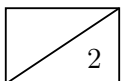
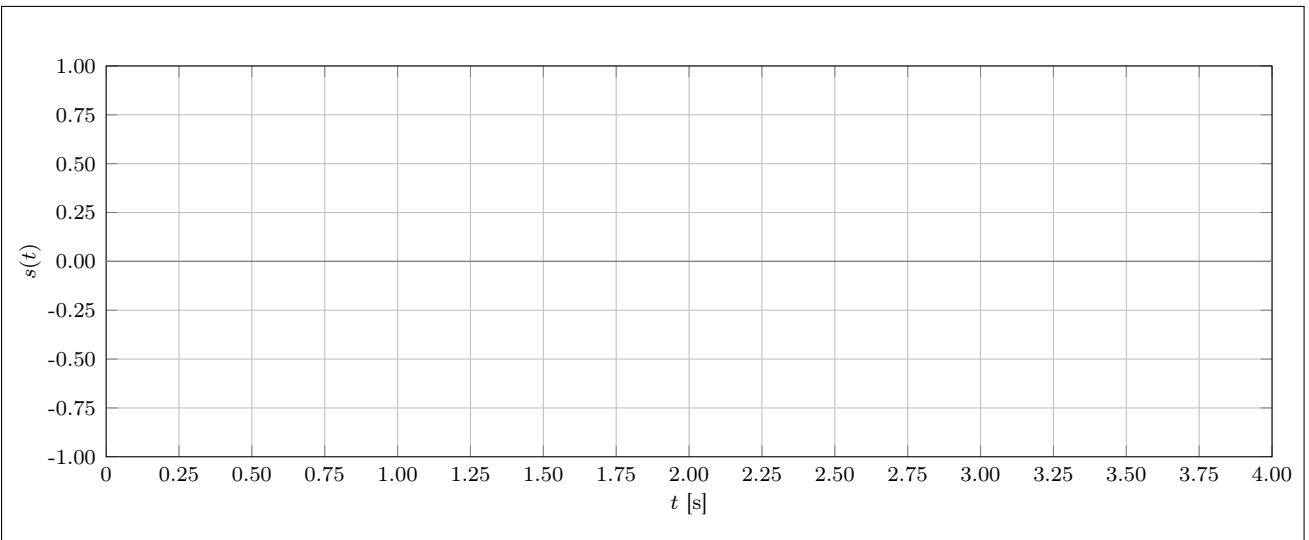


g)* Geben Sie die Quantisierungsstufen als Zahlenwerte an. Tragen Sie diese direkt in Abbildung 1.2a unterhalb der jeweiligen Codewörter ein.



h) Zeichnen Sie das zeit- und wertdiskrete Signal in das untenstehende Koordinatensystem.

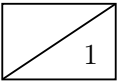
Hinweis: Signalamplituden $|s(t)| < 0.1$ werden als Ruhepotential interpretiert und auf den Wert 0 abgebildet.



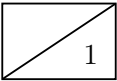
i) Geben Sie die empfangene Bitfolge an.

3 Name: _____

j)* Begründen Sie, ob es sich bei Return-to-Zero um einen gleichstromfreien Leitungscode handelt.

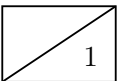


k)* Begründen Sie, ob bei Verwendung von Return-to-Zero eine automatische Taktrückgewinnung möglich ist.

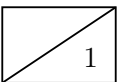


Die empfangene Bitfolge sei durch einen sehr einfachen Kanalcode geschützt. Dazu wurde vom Sender an eine Gruppe von je 3 bit ein sogenanntes Paritätsbit angehängt, welches 0 ist, wenn die vorangehenden 3 bit eine gerade Anzahl an Einsen enthält. Andernfalls ist das Paritätsbit 1.

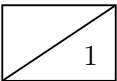
l)* Begründen Sie, welche Bitfehler mittels des Paritätsbits erkannt werden können.



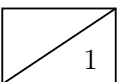
m)* Begründen Sie, welche Bitfehler mittels des Paritätsbits korrigiert werden können.

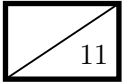


n) Markieren Sie die Paritätsbits in Ihrer Lösung von Teilaufgabe i).



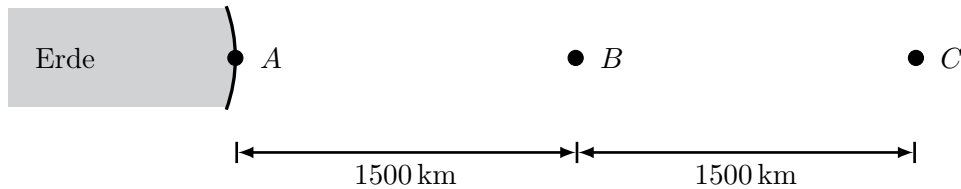
o) Prüfen Sie die empfangene Bitfolge auf Übertragungsfehler.





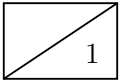
Aufgabe 2 Zeitsynchronisation (11 Punkte)

Ein hypothetischer Pico-Satellit B befindet sich in einer niedrigen Erdumlaufbahn. Wir betrachten den Zeitpunkt, zu dem er eine Entfernung von 1500 km zur Bodenstation A auf der Erdoberfläche hat. Er befindet sich damit direkt zwischen der Bodenstation und einem weiteren, baugleichen Satelliten C in einem höheren Orbit, der eine Entfernung von 3000 km zur Bodenstation hat.

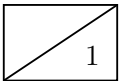


Die Modems der Satelliten können mit 14.4 kbit/s senden und mit 56.0 kbit/s empfangen. Die Bodenstation sendet und empfängt mit den maximalen Datenraten, die die Satelliten unterstützen. Nehmen Sie als Lichtgeschwindigkeit auf allen Strecken $c \approx 3 \cdot 10^8$ m/s an.

Satellit B möchte seine Borduhr mit der Uhrzeit der Bodenstation synchronisieren. Dazu wird folgendes, einfaches Protokoll durchgeführt: B sendet eine 64 B lange Anfrage an die Bodenstation. Diese antwortet mit einem 64 B langen Rahmen, der die aktuelle Uhrzeit der Bodenstation enthält. Verarbeitungszeiten an den Endpunkten sollen in dieser Aufgabe vernachlässigt werden.

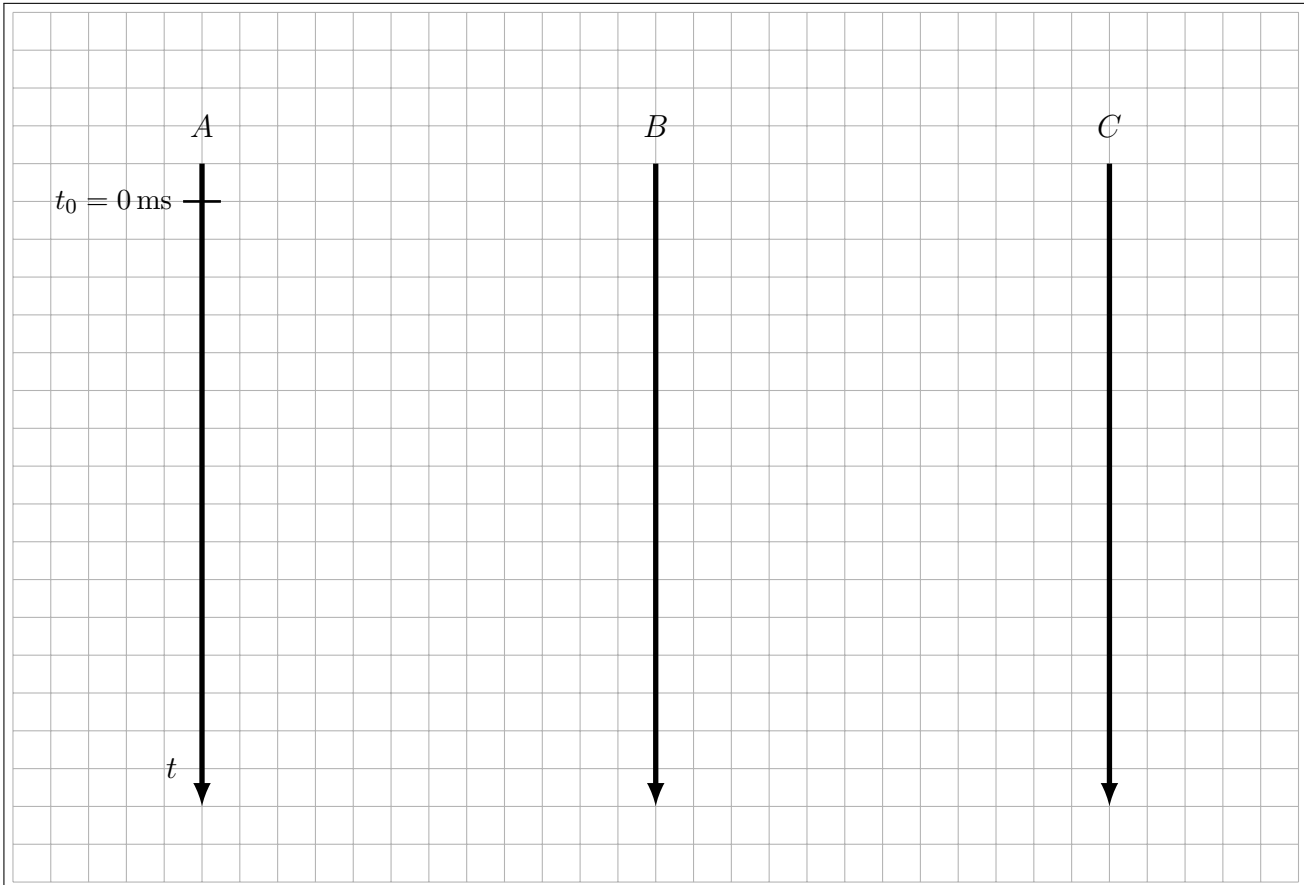
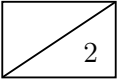


a)* Bestimmen Sie die Serialisierungszeiten $t_s(A, B)$ und $t_s(B, A)$ zwischen der Bodenstation A und dem Satelliten B .

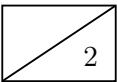


b)* Bestimmen Sie die Ausbreitungsverzögerung $t_p(A, B)$ zwischen der Bodenstation A und dem Satelliten B .

c) Zeichnen Sie die Anfrage von Satellit B und die Antwort der Bodenstation A in das folgende Weg-Zeit-Diagramm ein. Die Anfrage wird zum Zeitpunkt $t_0 = 0$ ms gesendet.
(Maßstab: horizontal 1 cm $\hat{=}$ 250 km, vertikal 1 cm $\hat{=}$ 10 ms)

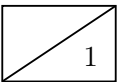


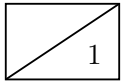
d) Kennzeichnen Sie die Serialisierungszeiten $t_s(A, B)$ und $t_s(B, A)$ sowie die Ausbreitungsverzögerung $t_p(A, B)$ und die Round-Trip-Time (RTT) in ihrer Lösung von Teilaufgabe c).



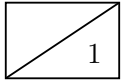
Durch die Serialisierungs- und Ausbreitungsverzögerungen stimmt die von der Bodenstation gesendete Zeit zum Zeitpunkt des Empfangs am Satelliten B nicht mehr exakt. Der Satellit schätzt die tatsächliche Uhrzeit, indem er die halbe RTT auf die empfangene Zeit aufaddiert. Hierbei entsteht ein Fehler Δt , der im Folgenden bestimmt werden soll.

e) Berechnen Sie die RTT aus Sicht des Satelliten B .

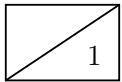




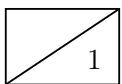
f)* Begründen Sie, warum hier nicht die korrekte Zeit berechnet wird, wenn man die halbe RTT auf die empfangene Zeit addiert.



g) Beim beschriebenen Verfahren addiert der Satellit die Korrektur $t_{c,S}$ auf die empfangene Zeit. Berechnen Sie $t_{c,S}$.



h) Berechnen Sie den korrekten Korrekturwert t_c , den der Satellit verwenden müsste, um die aktuelle Zeit zu bestimmen.



i) Berechnen Sie den absoluten Fehler Δt , den der Satellit beim beschriebenen Verfahren macht.

Aufgabe 3 NAT und statisches Routing (15 Punkte)

15

Gegeben sei die Netztopologie aus Abbildung 3.1. PC1 und PC2 sind über ein gewöhnliches Ethernet-Switch SW1 mit Router R1 verbunden, welches Zugang zum Internet ermöglicht.

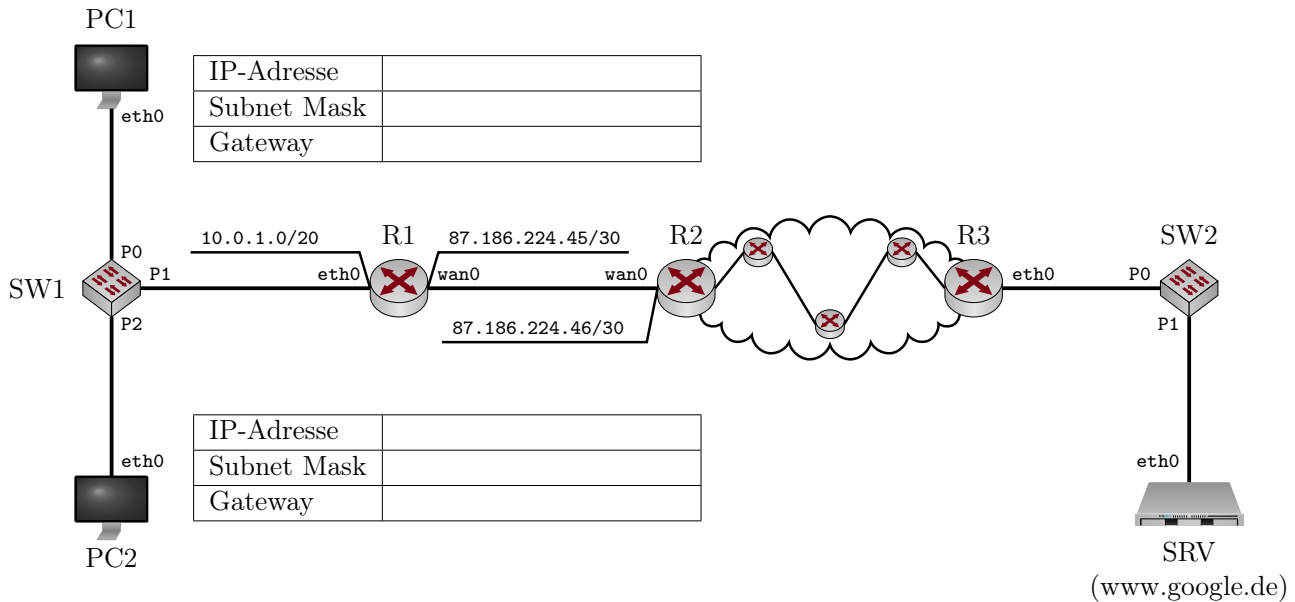


Abbildung 3.1: Netztopologie

a)* Begründen Sie, ob die Adresse 10.0.1.0 für das gegebene Präfix nutzbar ist. Falls nein, vergeben Sie an R1 eine sinnvolle Adresse im selben Netz.

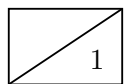
1

b)* Bestimmen Sie die Netzadresse und Broadcastadresse des Netzwerks, in dem sich PC1, PC2 und R1 befinden.

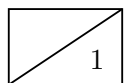
1

c)* Wieviele IP-Adressen stehen in diesem Netzwerk zur Adressierung von Geräten zur Verfügung?

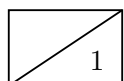
1



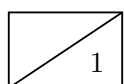
d) Weisen Sie PC1 und PC2 jeweils eine sinnvolle IP-Adresse, Subnetzmaske und Gateway-Adresse zu, so dass diese eine Verbindung zum Internet herstellen können. Tragen Sie die Werte direkt in Abbildung 3.1 ein.



e)* Wie viele /20 Subnetze gibt es im Netz 10.0.0.0/8?

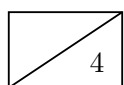


f)* Begründen Sie, weswegen R1 NAT unterstützen muss, damit PC1 und PC2 mit Hosts im Internet kommunizieren können.

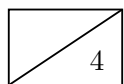


g)* Welche Informationen muss R1 mind. in seiner NAT-Tabelle vorhalten?

Im Folgenden kürzen wir IP- und MAC-Adressen nach dem Schema `<Gerätename>.<Interface>` ab, z. B. `R1.wan0`. Beachten Sie zur Bearbeitung der beiden folgenden Teilaufgaben außerdem, dass sich zwischen R2 und R3 insgesamt drei weitere Router befinden. PC1 greift nun auf die Webseite <http://www.google.de> zu.



h) Ergänzen Sie für die Anfrage von PC1 an www.google.de die Headerfelder in den drei leeren Kästen in Abbildung 3.2. Sofern ein Feld nicht eindeutig bestimmt ist, treffen Sie eine sinnvolle Wahl.



i) Ergänzen Sie für die Antwort von www.google.de an PC1 die Headerfelder in den drei leeren Kästen in Abbildung 3.3. Sofern ein Feld nicht eindeutig bestimmt ist, treffen Sie eine sinnvolle Wahl.

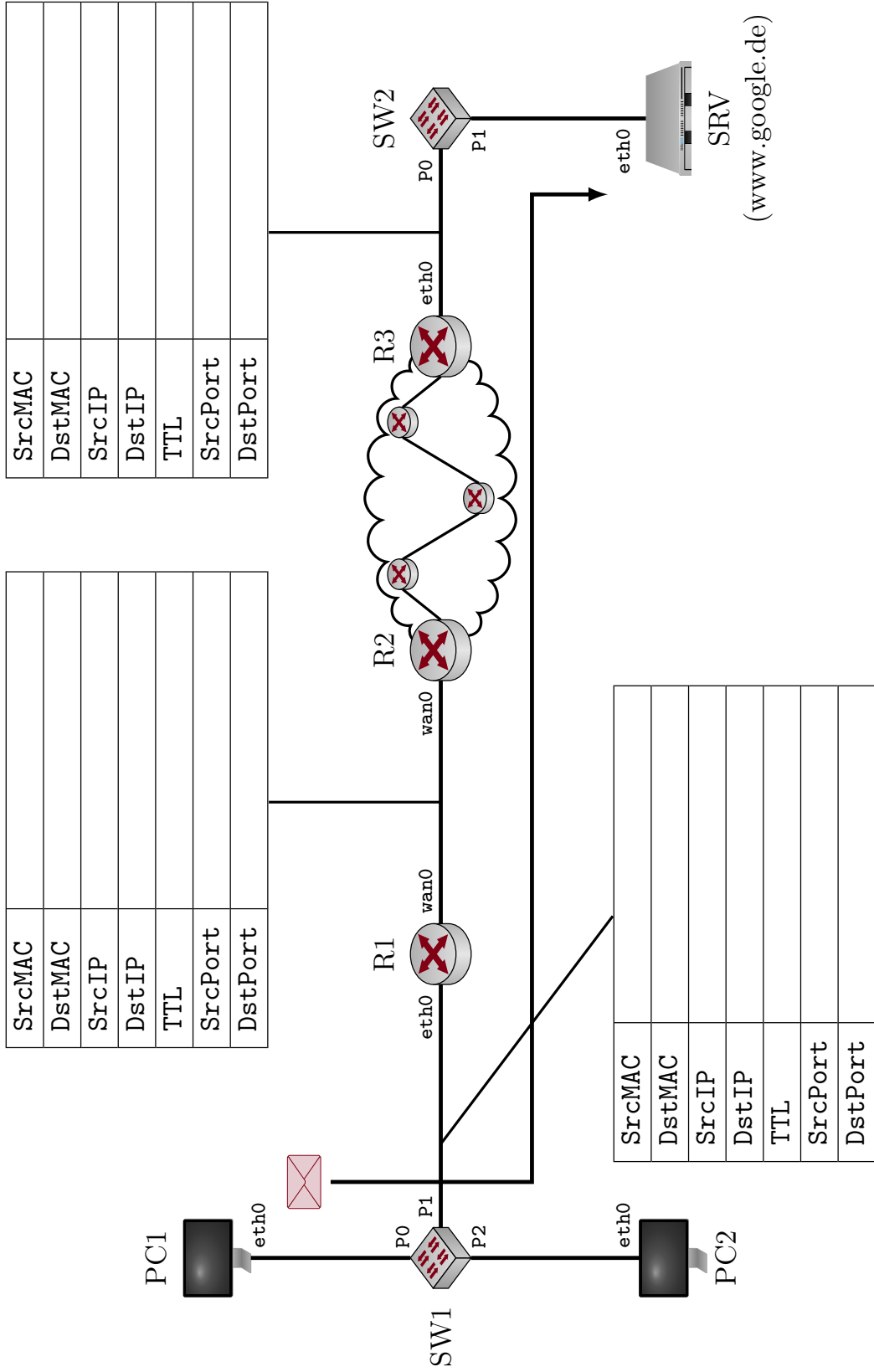


Abbildung 3.2: Netztopologie

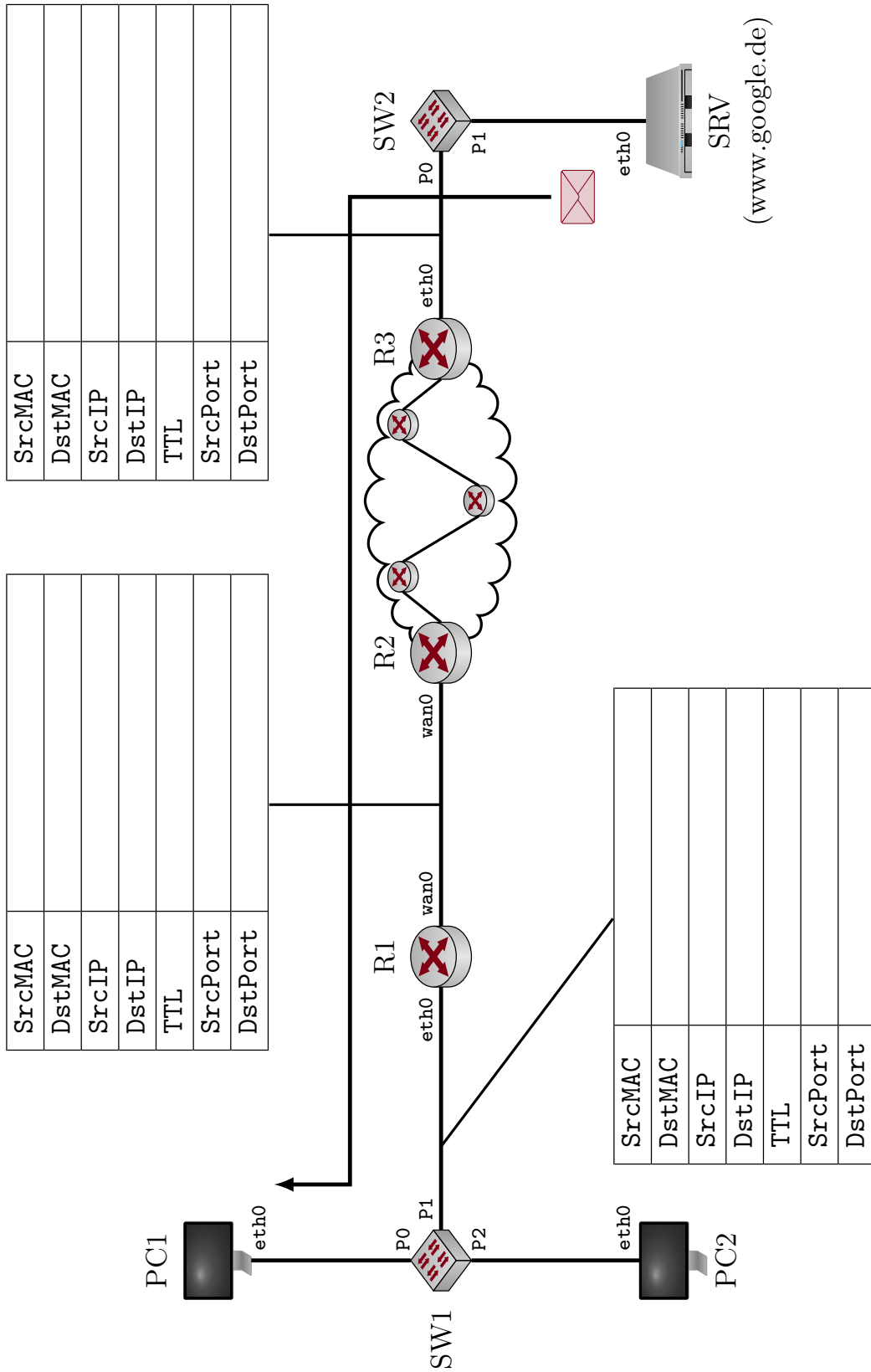


Abbildung 3.3: Netztopologie

Aufgabe 4 Wireshark (11 Punkte)

11

Gegeben sei das Netzwerk aus Abbildung 4.1. Von PC1 aus wird ein Paket in Richtung PC2 gesendet. Der Datenverkehr wird von PC1 mittels Wireshark aufgenommen. Der dekodierte Trace ist in Tabelle 1 abgebildet. Leider sind die Timestamps ein wenig durcheinander geraten – weswegen wir sie gar nicht erst abgedruckt haben. Die Rahmen sind also **nicht** in der Reihenfolge dargestellt, wie sie von PC1 gesehen wurden. Dieser Trace soll in den folgenden Teilaufgaben analysiert und in die richtige Reihenfolge gebracht werden.

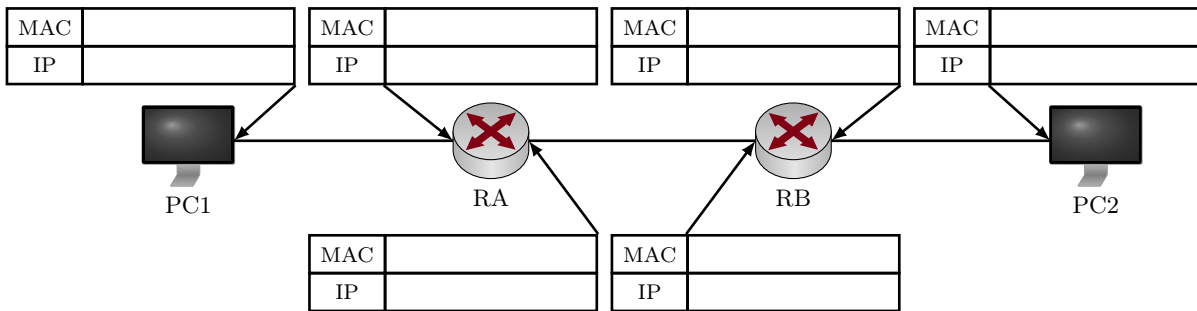


Abbildung 4.1: Netztopologie

a)* Erklären Sie zunächst den Unterschied zwischen MAC- und IP-Adressen hinsichtlich ihrer Verwendung.

1

Wie eingangs erwähnt sind die Rahmen in Tabelle 1 **nicht** in der richtigen Reihenfolge. Die Identifier A – H in der ersten Spalte dienen in den nachfolgenden Teilaufgaben dazu, auf einzelne Rahmen zu verweisen.

b)* Erläutern Sie kurz, wozu die beiden Rahmen B und G dienen.

1

Hinweis: Die MAC-Adresse `33:33:ff:00:00:01` in Rahmen G kann in dieser Aufgabe als Broadcast-adresse (`ff:ff:ff:ff:ff:ff`) betrachtet werden.

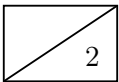
c)* Erläutern Sie, wie es zu Rahmen H kommen kann.

1

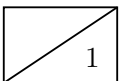
Id.	in	out	Nr.	Output
A				Ethernet II, 90:e2:ba:2a:d2:a3 -> 90:e2:ba:2a:8d:0f, Length: 1294 fc00::10 -> fc00::2:10, IPv6, Length: 1294 IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0x7c0d6306)
B				Ethernet II, 90:e2:ba:2a:8d:0f -> 90:e2:ba:2a:d2:a3, Type: IPv6 (0x86dd) fc00::1 -> fc00::10, ICMPv6, Lenth: 86 Neighbor Advertisement: fc00::1 (rtr, sol, ovr) is at 90:e2:ba:2a:8d:0f
C				Ethernet II, 90:e2:ba:2a:8d:0f -> 90:e2:ba:2a:d2:a3, Length: 238 fc00::2:10 -> fc00::10, ICMPv6, Length: 238 IPv6 fragment (nxt=ICMPv6 (58) off=154 id=0x7aef85b4) Echo (ping) reply id=0x1dd7, seq=2
D				Ethernet II, 90:e2:ba:2a:d2:a3 -> 90:e2:ba:2a:8d:0f, Type: IPv6 (0x86dd) fc00::10 -> fc00::2:10, ICMPv6, Length: 1462 Echo (ping) request id=0x1dd7, seq=1
E				Ethernet II, 90:e2:ba:2a:d2:a3 -> 90:e2:ba:2a:8d:0f, Length: 238 fc00::10 -> fc00::2:10, ICMPv6, Length: 238 IPv6 fragment (nxt=ICMPv6 (58) off=154 id=0x7c0d6306) Echo (ping) request id=0x1dd7, seq=2
F				Ethernet II, 90:e2:ba:2a:8d:0f -> 90:e2:ba:2a:d2:a3, Length: 1294 fc00::2:10 -> fc00::10, IPv6, Length: 1294 IPv6 fragment (nxt=ICMPv6 (58) off=0 id=0x7aef85b4)
G				Ethernet II, 90:e2:ba:2a:d2:a3 -> 33:33:ff:00:00:01, Type: IPv6 (0x86dd) fc00::10 -> ff02::1:ff00:1, ICMPv6, Length: 86 Neighbor Solicitation: whois fc00::1 from 90:e2:ba:2a:d2:a3
H				Ethernet II, 90:e2:ba:2a:8d:0f -> 90:e2:ba:2a:d2:a3, Type: IPv6 (0x86dd) fc00::1:2 -> fc00::10, ICMPv6, Length: 1294 Packet Too Big, MTU: 1280

Tabelle 1: Mitschnitt des Netzwerkverkehrs an PC1 in unsortierter Reihenfolge.

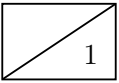
d) Erläutern Sie den Zusammenhang zwischen den Rahmen A, D und E



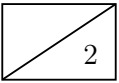
e)* Bestimmen Sie die MTU auf dem Link zwischen RB und PC2.



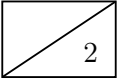
f)* Bestimmen Sie für alle Rahmen aus Tabelle 1, ob es sich von PC1 aus gesehen um eingehende oder ausgehende Rahmen handelt. Kreuzen Sie dazu das entsprechende Feld in den Spalten „in“ bzw. „out“ in Tabelle 1 an.



g) Bringen Sie nun die Rahmen aus Tabelle 1 in eine sinnvolle Reihenfolge. Notieren Sie dazu in der Spalte „Nr.“ die Nummer des Pakets beginnend bei 1.

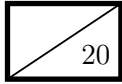


h) Tragen Sie nun – soweit dies aus Tabelle 1 hervorgeht – alle MAC- und IP-Adressen der Geräte in Abbildung 4.1 ein. Sofern ein Eintrag aus Tabelle 1 **nicht** hervorgeht, **streichen Sie das zugehörige Feld deutlich**.



Hinweis: Die MAC-Adresse `33:33:ff:00:00:01` in Rahmen G kann in dieser Aufgabe als Broadcast-adresse (`ff:ff:ff:ff:ff:ff`) betrachtet werden¹.

¹Es handelt sich um eine sog. Multicast-Adresse, mit der eine Gruppe von Stationen innerhalb der lokalen Broadcast-Domäne adressiert werden. Das spielt für diese Aufgabe aber keine Rolle.



Aufgabe 5 Datenkompression (20 Punkte)

In dieser Aufgabe betrachten wir eine vereinfachte Version des ITU T.30-Protokolls, besser bekannt als Telefax („Fax“). Abbildung 5.1 stellt einen 9×50 Pixel großen Ausschnitt einer Seite dar, welche per Fax übertragen werden soll.

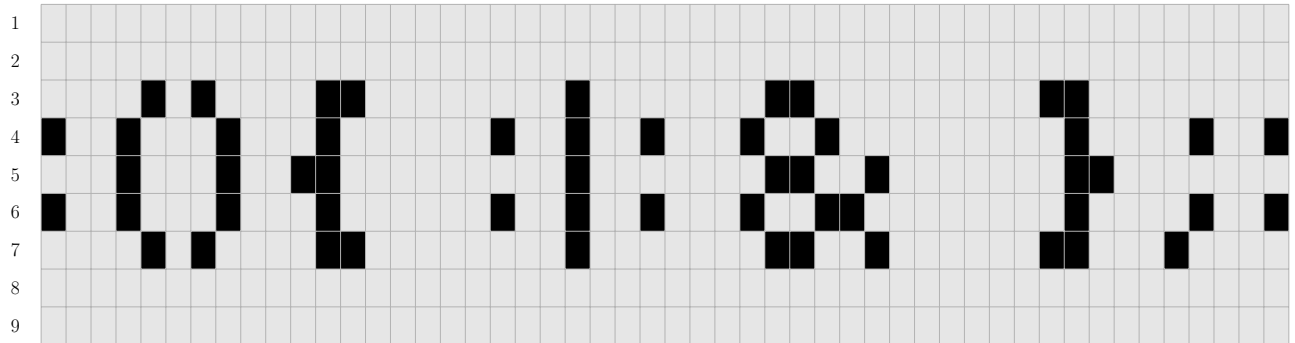
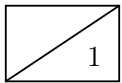


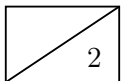
Abbildung 5.1: Ausschnitt einer schwarz/weißen Telefaxseite. Die Zahlen am linken Rand geben die Zeilennummer an.

Lösungsfeld für Teilaufgabe f)

Eine einfache, aber ineffiziente, Möglichkeit besteht darin, schwarze Pixel durch eine logische 0 und weiße Pixel durch eine logische 1 zu kodieren. Wir bezeichnen diese Art der Kodierung im Folgenden als *einfachen Code*.

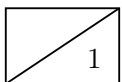


a)* Bestimmen Sie die Länge der zu übertragenden Daten in Bit, wenn der Seitenausschnitt mit diesem einfachen Code kodiert wird.



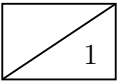
b)* Bestimmen Sie den Informationsgehalt der beiden verwendeten Codewörter.

Hinweis: Der Seitenausschnitt besteht aus 55 schwarzen Pixeln.



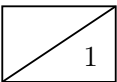
c) Bestimmen Sie die Entropie des Seitenausschnitts.

d) Was lässt sich aus dem Ergebnis von Teilaufgabe (c) folgern?

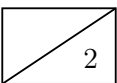


Zur verlustfreien Kompression verwendet ITU T.30 eine Kombination aus Lauflängenkodierung (engl. Run Length Encoding, RLE) und anschließender Huffman-Kodierung. Hierzu werden zunächst zeilenweise die Anzahl gleichfarbiger aufeinanderfolgender Pixel zusammen mit dem jeweiligen Farbwert (schwarz oder weiß) kodiert, beispielsweise **3w** bzw. **4s** für drei aufeinanderfolgende weiße bzw. vier aufeinanderfolgende schwarze Pixel.

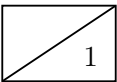
e)* Erläutern Sie, was man unter verlustfreier Kompression versteht.



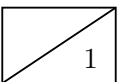
f)* Geben Sie die **dritte** Zeile des Seitenausschnitts in Abbildung 5.1 lauffängenkodiert im Lösungsfeld unterhalb von Abbildung 5.1 an.



g)* Der Huffman-Code ist ein sog. präfixfreier Code. Erläutern Sie, was man darunter versteht.



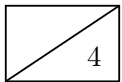
h) Inwiefern erleichtert die Verwendung präfixfreier Codes die Dekompression?



Über den gesamten Seitenausschnitt ergeben sich die in Tabelle 2 aufgelisteten RLE-Codewörter mit ihren entsprechenden absoluten Häufigkeiten. Die Gesamtzahl der RLE-Codewörter für den kompletten Ausschnitt beträgt 95.

RLE-Wort	Häufigkeit	Huffman-Codewort
1s	35	
2w	13	
2s	10	
3w	9	
4w	7	
7w	5	
8w	4	
50w	4	
9w	3	
6w	3	
1w	2	
—	$\sum = 95$	—

Tabelle 2: RLE-Codewörter, sortiert nach Häufigkeit im Seitenausschnitt aus Abbildung 5.1.



i)* Konstruieren Sie aus den Angaben in Tabelle 2 einen Huffman-Code. Beginnen Sie dazu mit den im Lösungsfeld abgedruckten Blättern des Baums.

2/95

1w

3/95

6w

3/95

9w

4/95

50w

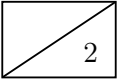
4/95

8w

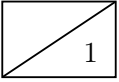
5/95

7w

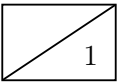
j) Geben Sie die Codewörter des von Ihnen konstruierten Codes an. Tragen Sie diese in Tabelle 2 ein.



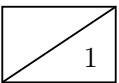
k) Bestimmen Sie die Gesamtlänge des komprimierten Seitenausschnitts in Bit.



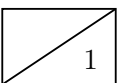
l) Um wieviel Prozent ist der komprimierte Seitenausschnitt kürzer im Vergleich zur einfachen Kodierung?

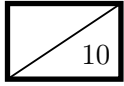


m)* Erläutern Sie, welche Information (abgesehen von den komprimierten Daten) auf der Empfängerseite zur Dekodierung noch benötigt wird.

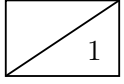


n) Geben Sie zwei Möglichkeiten an, wie das Problem aus Teilaufgabe m) in der Praxis gelöst werden kann.

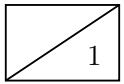


**Aufgabe 6 Kurzaufgaben (10 Punkte)**

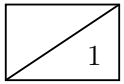
Die folgenden Kurzaufgaben sind **jeweils unabhängig voneinander**. Stichpunktartige Antworten sind ausreichend!



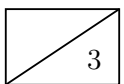
a)* Begründen Sie, ob mit Hilfe einer CRC-Checksumme, welche aus dem Generatorpolynom $g(x) = x^5 + x^2 + 1$ erzeugt wurde, Burstfehler der Länge 4 erkannt werden.



b)* Geben Sie die Integerzahl 240 als 2 B langen Hexadezimalwert in Little Endian und Network Byte Order an.

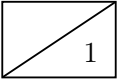


c)* Nennen Sie zwei verschiedene Gründe, warum man einen Dienst als verteiltes System anstatt eines einzelnen, zentralen Servers implementiert.

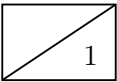


d)* Beschreiben Sie in 3-4 Stichpunkten, wie das Programm `traceroute` funktioniert.

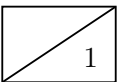
e)* Was sind zwei grundlegende Unterschiede zwischen IPv4 und IPv6?



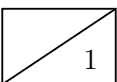
f)* Erklären Sie, warum ein Programm, das auf 10 Rechnern parallel anstatt auf einem ausgeführt wird, weniger als 10 mal so schnell sein kann.



g)* Erläutern Sie das Konsistenzproblem, das sich ergibt, wenn mehrere Knoten einer verteilten Anwendung auf gemeinsam genutzten Speicher zugreifen.



h)* Ein Angreifer möchte Daten in eine bestehende TCP-Verbindung einschleusen. Nennen Sie die Informationen über den TCP-Zustand, die er dafür benötigt.



Zusätzlicher Platz für Lösungen – bitte markieren Sie deutlich die Zugehörigkeit zur jeweiligen Aufgabe und streichen Sie ungültige Lösungen!

