Chair of Network Architectures and Services
Department of Informatics
Technical University of Munich

Eexam
Place student sticker here

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Grundlagen Rechnernetze und Verteilte Systeme

Exam:	IN0010 / Hausaufgabe 11	**Date:**	Monday 15th June, 2020
Examiner:	Prof. Dr.-Ing. Georg Carle	**Time:**	14:00 – 23:59

Working instructions

- Die erreichbare Gesamtpunktzahl beträgt 36 credits.
- Bitte geben Sie bis spätestens Montag, den **20. Juli um 23:59 CEST** über TUMexam ab.
  Bitte haben Sie Verständnis, wenn das Abgabesystem noch nicht reibungslos funktioniert. Wir arbeiten daran!
- Ihren **persönlichen** Link zur Abgabe finden Sie auf Moodle. Geben Sie diesen **nicht** weiter.
- Bitte haben Sie Verständnis, falls die Abgabeseite zeitweilig nicht erreichbar ist.

Bitte nehmen Sie die Hausaufgaben dennoch ernst:

- Neben der Einübung des Vorlesungsstoffs und der Klausurvorbereitung dienen die Hausaufgaben auch dazu, den Ablauf der Midterm zu erproben.
- Finden Sie einen für sich selbst praktikablen und effizienten Weg, die Hausaufgaben zu bearbeiten. Hinweise hierzu haben wir auf `https://grnvs.net.in.tum.de/homework_submission_details.pdf` für Sie zusammengestellt.

Left room from _____ to _____ / Early submission at _____

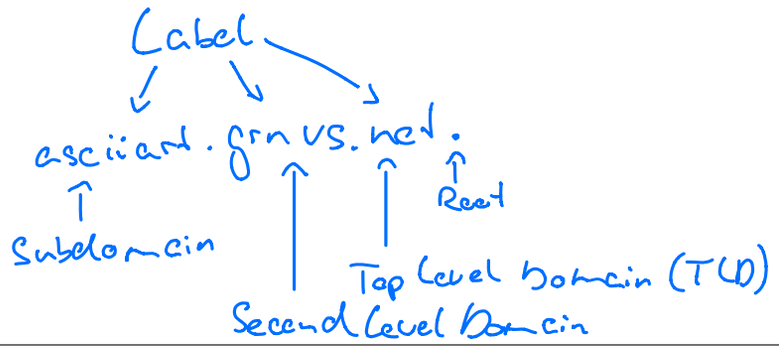
Problem 1 Domain Name System (DNS) (14 credits)

Hinweis: Angelehnt an Endterm 2015
Zentrale Aufgabe des Domain Name Systems (DNS) ist es, menschenlesbare Namen auf IP-Adressen abzubilden, die dann für die Wegwahl auf der Netzwerkschicht verwendet werden können. Bei dem Namen `asciiart.grnvs.net.` handelt es sich um einen sog. *Fully Qualified Domain Name (FQDN)*.

0 1 a)* Was ist der Unterschied zwischen einem vollqualifizierten Domain Name (FQDN) und einem nicht-(voll)qualifizierten?

- FQDN endet stets mit: . (Wurzel des Namespaces) www.grnvs.net.
- ein nicht (voll) qualifizierter ist dabei relativ zu einer anderen Wurzel www.grnvs

0 1 2 b)* Benennen Sie die einzelnen Bestandteile des FQDNs, sofern es dafür gängige Bezeichnungen gibt.



In Abbildung 1.1 sind ein PC sowie eine Reihe von Servern dargestellt. Wir nehmen an, dass PC1 den Router als Resolver nutzt. Der Router wiederrum nutzt einen Resolver von Google unter der IP-Adresse 8.8.8.8 zur Namensauflösung. Ferner nehmen wir an, dass der Google-Resolver gerade neu gestartet wurde (also insbesondere keine Resource Records gecached hat) und rekursive Namensauflösung anbietet.
Die autoritativen Nameserver für die jeweiligen Zonen sind in Tabelle 1.1 gegeben.

Zone	autoritativer Nameserver
`.`	`d.root-servers.net.`
`com.`, `net.`	`a.gtld-servers.net.`
`google.com.`	`ns1.google.com.`
`grnvs.net.`	`bifrost.grnvs.net.`

Table 1.1: Zonen mit zugehörigen autoritativen Nameservern

0 1 2 c)* Erläutern Sie den Unterschied zwischen einem *Resolver* und einem *Nameserver*.

Nameserver: Nameserver sind autoritativ für mehrere Zonen.
↳ Besitzen eine vollständige Kopie der autoritativen Zone.

Resolver: Resolver lösen einen Domain Namen mittels iterativer Anfragen an die Nameserver auf und geben das Ergebnis an den anfragenden Client zurück.

d)* Welche Funktion erfüllen `d.root-servers.net` und `a.gtld-servers.net`?

0
1

d.root-servers.net: Root-Nameserver, kennt alle Nameserver, welche für die TLDs verantwortlich sind.

a.gtld-servers.net: Kennt alle Nameserver unter den Second-Level-Domains .com, .net.

e)* Erklären Sie den Unterschied zwischen iterativer und rekursiver Namensauflösung.

0
1
2

iterativ: Für einzelne Zonen autoritative Nameserver werden vom Client selbst nacheinander angefragt.

rekursiv: Die DNS-Anfrage wird an einen Resolver weiter geleitet, welcher selbst iterative Namensauflösung betreibt und das Ergebnis an den Client zurück schickt.

f) Zeichnen Sie in Abbildung 1.1 alle DNS-Nachrichten (Requests / Responses) ein, die ausgetauscht werden, sobald PC1 auf `asciiart.grnvs.net.` zugreift. Nummerieren Sie die Nachrichten gemäß der Reihenfolge, in der sie zwischen den einzelnen Knoten ausgetauscht werden.

0
1
2
3
4
5

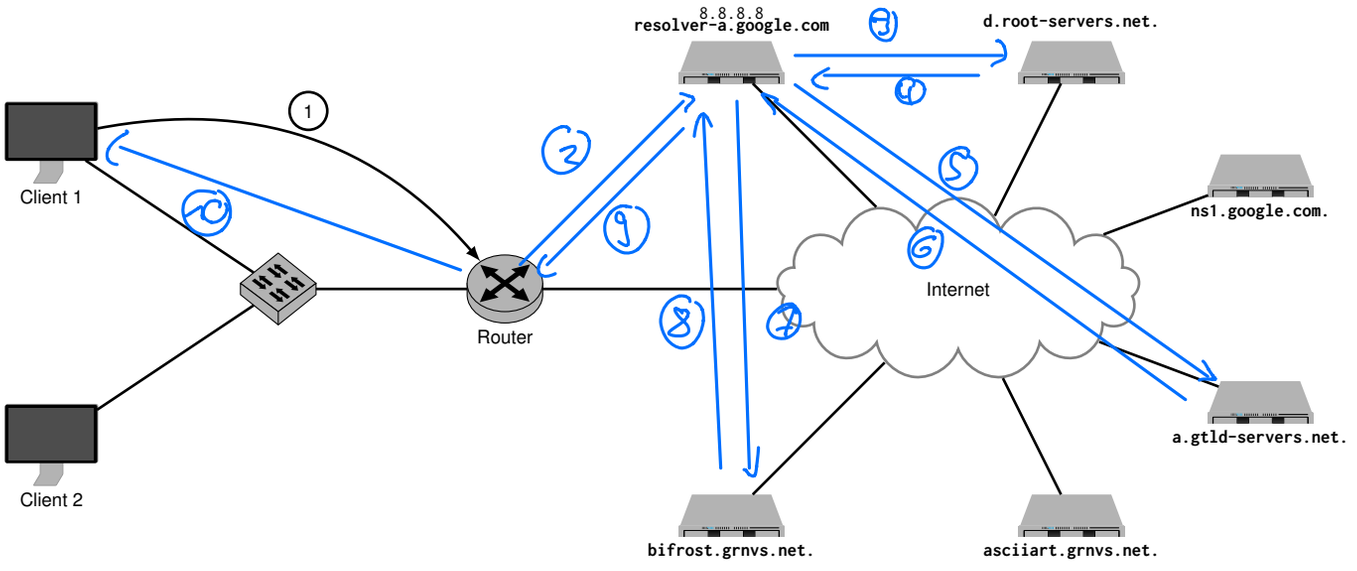


Figure 1.1: Vorlage zu Aufgabe 1f)

g)* Wie wird im DNS sichergestellt, dass kein bösartiger Nameserver Anfragen für andere Domänen beantwortet? (Wir gehen davon aus, dass keine Man-in-the-Middle-Angriffe möglich sind.)

0
1

Dies wird nur dadurch sichergestellt, dass ein Nameserver stets nur autoritative, ebenfalls vertraute Nameserver zum Resolver schickt.

Das hilft jedoch nicht gegen Man-in-the-Middle (MitM) Angriffe. Diese können jedoch mittels der DNSSEC Erweiterung verhindert werden.

Problem 2 **Kompression: Huffman-Kodierung (12 credits)**

Gegeben sei das Alphabet $\mathcal{A} = \{a, b, c, d\}$ und die Nachricht

$$m = \text{aabcbdacababbbbcbddbbbaababdbdbb} \in \mathcal{A}^{32}.$$

0 1 2 a)* Bestimmen Sie die Auftrittswahrscheinlichkeiten $p_i \in \mathcal{A}$ der einzelnen Zeichen in m.

Zeichen	p_i
a	$8/32 = 1/4$
b	$16/32 = 1/2$
c	$3/32 \approx 0.09$
d	$5/32 \approx 0.16$

0 1 2 b) Bestimmen Sie den Informationsgehalt $I(p_i)$ der einzelnen Zeichen aus $\mathcal{A}$.

$I(p_i) = -\log_2(p_i)$ bit

$I(p_a) = -\log_2(1/4) = 2$ bit

$I(p_b) = 1$ bit

$I(p_c) \approx 3{,}41$ bit

$I(p_d) \approx 2{,}7$ bit

0 1 2 c) Die Nachricht m stamme aus einer Nachrichtenquelle X. Bestimmen Sie auf Basis der bisherigen Ergebnisse die Quellenentropie $H(X)$.

$H(A) = \sum_{i \in A} p_i \, I(p_i)$

$H(X) = \sum_{i \in A} p_i \, I(p_i) = \frac{1}{4} \cdot 2 \text{ bit} + \frac{1}{2} \cdot 1 \text{ bit} + \frac{3}{32} \cdot 3{,}4 \text{ bit} + \frac{5}{32} \cdot 2{,}7 \text{ bit}$

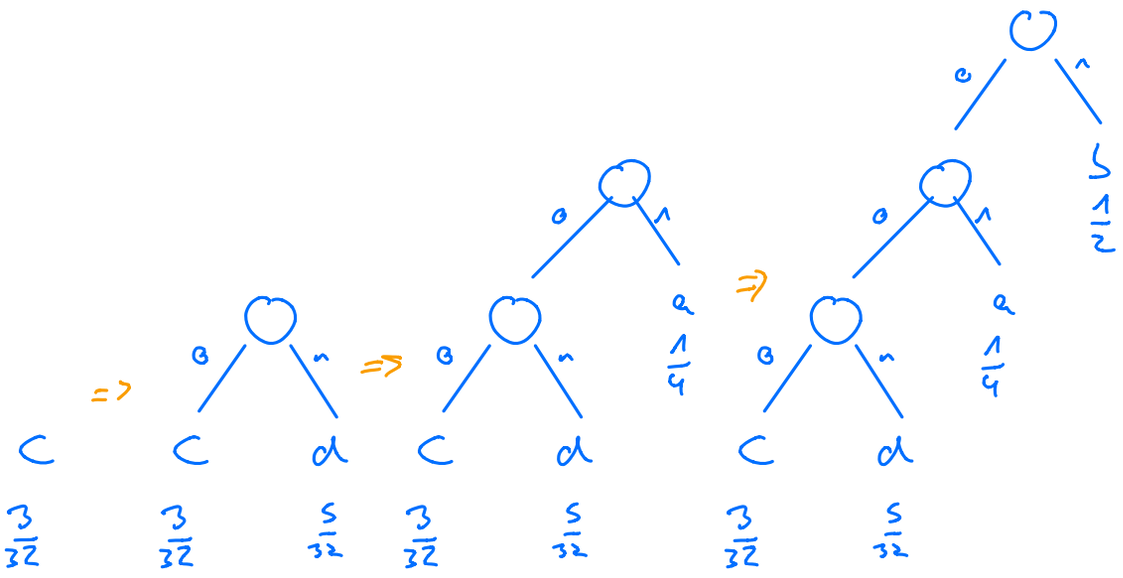
$= 1{,}738$ bit

Damit lässt sich jedes Zeichen der Quelle mit durchschnittlich 1,738 bit codieren.

d) Bestimmen Sie nun einen binären Huffman-Code C für diese Nachrichtenquelle.

0 1 2 3

Das Zeichen mit der jeweils kleinsten Auftrittswahrscheinlichkeit wird ganz unten im Baum eingefügt.
Dann wird für jede ausgehende Kante $\{0,1\}$ vergeben



$C = \{a \mapsto 01,\ b \mapsto 1,\ c \mapsto 000,\ d \mapsto 001\}$

e) Bestimmen Sie die durchschnittliche Codewortlänge von C.

0 1 2

Die durchschnittliche Codewortlänge ergibt sich aus der gewichteten Summe aller Codewortlängen multipliziert mit der entsprechenden Auftrittswahrscheinlichkeit.

$$\bar{l}_C = \sum_{i \in A} p_i \cdot l(c(i)) = \frac{1}{4} \cdot 2\,\text{bit} + \frac{1}{2} \cdot 1\,\text{bit} + \frac{5}{32} \cdot 3\,\text{bit} + \frac{3}{32} \cdot 3\,\text{bit}$$

$$= 1.738\,\text{bit}.$$

1.738 bit

f) Vergleichen Sie die durchschnittliche Codewortlänge von C mit der Codewortlänge eines uniformen[1] Binärcodes.

0 1

Der kürzeste uniforme Code hat Länge 2 (4 Zeichen).

Damit beträgt die Differenz zur optimalen Codewortlänge $\bar{l}_C$:

12,5 %

[1] Ein Code heißt *uniform*, wenn alle Codewörter dieselbe Länge aufweisen.

Problem 3 SMTP (Hausaufgabe) (10 credits)

Für daheim: Mailversand von der Kommandozeile
Sie haben in der Vorlesung gelernt, dass mithilfe des Programms `telnet` bzw. mit dem s_client von OpenSSL Verbindungen zu Webservern aufgebaut werden können.
Ihre Aufgabe ist nun, sich mithilfe des s_client mit dem SMTP-Servers Ihres Mailproviders zu verbinden und sich selbst eine e-Mail zu schicken. Die notwendigen Schritte können Sie sich aus dem RFC 5321 erschließen. Alternativ ist es auch möglich entsprechende Tutorials zu konsultieren. Der initiale Befehl könnte folgendermaßen aussehen: `openssl s_client -crlf -connect <smtp.server.org>:465`
Sollten Sie CRAM-MD5 verwenden wollen, könnte Ihnen die folgende Bash-Funktion zur Berechnung der Response auf eine vom Server gegebene Challenge helfen:

```
cram() {
  challenge=$1
  username=$2
  challenge=$(echo -n $challenge|base64 -d)
  echo "Challenge is: $challenge"
  read -sp "Password for $username: " password
  echo ""
  hash=$(echo -n "$challenge" |openssl md5 -hmac "$password" -hex|cut -d" " -f 2 )
  response=$(echo -n "$username $hash" |base64)
  echo "Response for server is: "
  echo $response
}
```

Rufen sie Sie dann folgendermaßen auf:
`cram <CHALLENGE> <USERNAME>`

0 1 2 3 4

a)* Pasten Sie den Output.

b) Ermitteln Sie, welche Authentifizierungsmethoden der von Ihnen gewählte SMTP-Server anbietet und erläutern Sie den Unterschied zwischen den Methoden.

0 1 2 3 4

c) Warum ist überhaupt eine Authentifizierung notwendig? Welche Probleme werden dadurch behoben oder zumindest verringert?

0 1 2

Additional space for solutions–clearly mark the (sub)problem your answers are related to and strike out invalid solutions.