

---

**Technische Informatik III:**  
**Betriebssysteme und Rechnernetze WS 2007/08**  
**Musterlösung zur Probeklausur**

---

**Aufgabe 1: Einfache Programmierung in C**

**10 Punkte**

Implementieren Sie in C die Funktion `int str_lowercase_compare(const char *s1, const char *s2)`, die überprüft, ob die beiden Strings identisch sind, ohne dabei zwischen Groß- und Kleinschreibung zu unterscheiden. Die Funktion gibt 1 zurück, falls sie beide Strings als gleich erkennt und 0 sonst.

Listing:

```
1 int str_lowercase_compare(const char *s1, const char *s2)
2 {
3     int result = 1;    /* Variable für das Ergebnis */
4     int i = 0;        /* Zähler */
5     int dist = 'a' - 'A'; /* Abstand zwischen den Zeichen */
6
7     /* Schleife bis zum ersten Auftreten von '\0' oder Ungleichheit */
8     while(s1[i] != '\0' && s2[i] != '\0' && result == 1)
9     {
10        /* Überprüfe auf Ungleichheit, sonst ist nichts zu tun */
11        if(s1[i] != s2[i])
12        {
13            /* Fall: s1[i] ist kleiner Buchstabe */
14            if(s1[i] <= 'z' && s1[i] >= 'a')
15            {
16                /* Überprüfe auf Ungleichheit*/
17                if(s1[i] != (s2[i] + dist) )
18                {
19                    result = 0;
20                }
21            }
22            /* Fall: s1[i] ist größer Buchstabe */
23            else if(s1[i] <= 'Z' && s1[i] >= 'A')
24            {
25                /* Überprüfe auf Ungleichheit*/
26                if(s1[i] != (s2[i] - dist) )
27                {
28                    result = 0;
29                }
30            }
31            /* Fall: s[i] ist sonstiges Zeichen */
32            else
33            {
34                result = 0;
35            }
36        }
37        ++i;
38    }
39
40    /* Gebe Ergebnis zurück */
41    return result;
42 }
```

**Aufgabe 2: Betriebssysteme**

**6 Punkte**

Welche der folgenden Aussagen treffen zu? (Punktabzug für falsche Antworten)

- a) Betriebssysteme können nur in hardwarenahen Sprachen wie Assembler und C programmiert werden.  
**Falsch**, es gibt Betriebssysteme die in Java geschrieben wurden.
- b) Ein Mikrokern stellt alleine noch kein Betriebssystem dar, da er typische Betriebssystemdienste nicht bereitstellt.  
**Richtig**, ein Mikrokern stellt zwar grundlegende Funktionen wie Speicher- und Prozessverwaltung bereit, aber implementiert keine Betriebssystemdienste.
- c) Scheduler haben schon in Time-Sharing-Systemen aktive Prozesse schlafen gelegt, um anderen Prozessen den Zugriff auf die Betriebsmittel zu erlauben.  
**Richtig**, in Time-Sharing-Systemen werden präemptive (prozessunterbrechende) Scheduling-Algorithmen eingesetzt um Paralleltät/Multitasking zu simulieren.

- d) Batchsysteme konnten zwar mehrere Task erhalten, diese wurden aber nur streng nacheinander ausgeführt.  
**Richtig**, Batchsysteme (Stapelsysteme) haben keine ausgeklügelte Prozessverwaltung und können die Task nur streng nacheinander ausführen.
- e) In Linux kann ein Fehler im Mousedriver das ganze System zum Absturz bringen, da dieser im Kernelmode läuft.  
**Richtig**, da Linux einen monolithischen Kernel hat.
- f) Die Interprozesskommunikation ist in monolithischen Systemen komplexer als in Systemen mit Mikrokern, da Mikrokern generell ein einfacheres Design besitzen.  
**Falsch**, im Mikrokern findet, im Gegensatz zum monolithischen Kernel, während der Kommunikation immer ein Kontextwechsel statt.
- g) Kernel- und Usermode-Trennung entspricht einem Protection-Ring-Design mit zwei Ringen.  
**Richtig**, in diesem Fall läuft der Kernel und Kernelmodule im Ring 0 und alle anderen Prozesse im Ring 1.
- h) Es existieren sowohl Unixsysteme, die auf monolithischem Design beruhen als auch welche die auf einem Mikrokern aufbauen.  
**Richtig**, z. B. hat Minix einen Mikrokern und Linux einen monolithischen Kernel.

### Aufgabe 3: physikalische Schicht

6 Punkte

Welche der folgenden Aussagen treffen zu? (Punktabzug für falsche Antworten)

- a) Im TCP/IP-Referenzmodell bildet die physikalische Schicht zusammen mit der Bitsicherungsschicht die unterste Ebene.  
**Richtig**, z. B. stellt Ethernet so eine unterste Ebene dar.
- b) Die Manchesterkodierung ist Teil der physikalischen Schicht.  
**Richtig**.
- c) Framing ist Teil der physikalischen Schicht.  
**Falsch**, sie ist Teil der Sicherungsschicht (Schicht 2)
- d) Auch in der physikalischen Schicht gibt es mit MAC-Adressen schon Routinginformationen.  
**Falsch**, MAC-Adressen sind Teil der Sicherungsschicht (Schicht 2).
- e) Der Wechsel des Übertragungsmediums impliziert den Wechsel der untersten Schicht im OSI-Modell.  
**Richtig**, z. B. der Wechsel von Kupfer auf Glasfaser erfordert eine andere Schicht 1.
- f) Fällt das verwendete physikalische Medium aus, hat das auch immer zu einem Abbruch der es benutzenden Anwendung zu Folge.  
**Falsch**, intelligente Anwendung erkennen den Ausfall reagieren entsprechend.
- g) Die Framing-Einsfolgen eines Datenstroms werden von der Manchesterkodierung nicht angerührt, damit das Framing intakt bleibt.  
**Falsch**, die Framing-Einsfolgen werden bei der Manchesterkodierung mitkodiert damit der Gleichspannungsanteil im Mittel auf Null bleibt.
- h) Das Vakuum des Alls kann ebenfalls Teil einer physikalischen Schicht sein.  
**Richtig**, im Falle von Übertragung mittels Funkwellen ist das Vakuum des Alls Teil der physikalischen Schicht.

### Aufgabe 4: Fehlerkorrektur

14 Punkte

Es sollen die Pakete 11101100, 00010100 und 11111111 übertragen werden.

- a) Berechnen Sie ein viertes Paket zum Zwecke der Vorwärtsfehlerkorrektur mit XOR-Methode. (2 Punkte)

Paket 1: 11101100

Paket 2: 00010100

Paket 3: 11111111

Paket 4: 00000111

- b) Nehmen Sie an, im dritten Paket (11111111) ist das dritte Bit gekippt. Kann der Empfänger den Fehler erkennen? Führen Sie die Rechnung aus. (2 Punkte)

|            |          |
|------------|----------|
| Paket 1:   | 11101100 |
| Paket 2:   | 00010100 |
| Paket 3:   | 11011111 |
| Paket 4:   | 00000111 |
|            |          |
| berechnet: | 00100000 |

Fehler beim dritten Zeichen zwar erkannt, kann aber nicht korrigiert werden, da man nicht weiß, im welchem Paket das Bit gekippt ist.

- c) Nehmen Sie an das zweite Paket (00010100) ist verloren gegangen. Zeigen Sie, wie der Empfänger aus den verbliebenen drei Paketen den Inhalt des zweiten Pakets rekonstruieren kann. (2 Punkte)  
 Angenommen es ist kein Bit von den Paketen 1, 3 und 4 gekippt, dann kann man das  $i$ -te Bit von Paket 2 durch die die Summe der  $i$ -ten Bits der anderen Pakete modulo 2 (entspricht XOR mit 3 Operanten) berechnen:

|                |          |
|----------------|----------|
| Paket 1:       | 11101100 |
| Paket 3:       | 11111111 |
| Paket 4:       | 00000111 |
|                |          |
| rekonstruiert: | 00010100 |

- d) Begründen Sie für den allgemeinen Fall mit Paketen P1, P2, P3 und P4 warum die zuvor durchgeführte Rechnung immer zum korrekten Inhalt des verloren Paketes führen muss. (8 Punkte)  
 Voraussetzung es tritt keinen Bitfehler auf.

- 1. Version:  
 Voraussetzung es tritt keinen Bitfehler auf.  
 Sei nun  $X[i]$  ist das  $i$ -te Bit von Paket  $X$ , dann gilt

$$(P1[i] + P2[i] + P3[i]) \mod 2 = P4[i] \quad (1)$$

für alle  $i$  von 1 bis zur Gesamtlänge eines Paketes.

Geht nun das Paket P4 verloren, dann kann man dass durch (1) bitweise wiederherstellen.

Geht nun o. B. d. A. Paket P1 verloren, dann kann man (1) umstellen und erhält:

$$P1[i] = (P4[i] - P2[i] - P3[i]) \mod 2 [= (P2[i] + P3[i] + P4[i]) \mod 2] \quad (2)$$

Mit (2) kann man nun das Paket P1 wieder bitweise wiederherstellen.

- 2. Version (die Bessere):

$$P4 = P1 \text{ xor } P2 \text{ xor } P3 \quad (3)$$

$$\begin{aligned}
 P1 \text{ xor } P3 \text{ xor } P4 &= P1 \text{ xor } P3 \text{ xor } (P1 \text{ xor } P2 \text{ xor } P3) && \text{, wegen (3)} \\
 &= (P1 \text{ xor } P1) \text{ xor } (P3 \text{ xor } P3) \text{ xor } P2 && \text{, da xor assoziativ und kommutativ} \\
 &= 0 \text{ xor } 0 \text{ xor } P2 && \text{, da } a \text{ xor } a = 0 \\
 &= P2 && \text{da } 0 \text{ xor } a = a
 \end{aligned}$$

## Aufgabe 5: Scheduling

**22 Punkte**

Vergleichen Sie die Scheduling-Algorithmen First Come First Serve (FCFS), Shortest Process Next (SPN), Shortest Remaining Time (SRT) und Highest Response Ratio Next (HRRN) im Bezug auf folgende Prozesse:

| Prozess | Ankunftszeit | Bearbeitungszeit |
|---------|--------------|------------------|
| A       | 0            | 5                |
| B       | 1            | 3                |
| C       | 3            | 2                |
| D       | 5            | 2                |
| E       | 6            | 1                |

- a) Erstellen für jeden der Algorithmen ein Zeitdiagramm. Sollten mehrere Prozesse die selbe Priorität besitzen, so wählen Sie denjenigen mit der geringsten Ankunftszeit. (16 Punkte)

- First Come First Server:

|   |           |         |       |
|---|-----------|---------|-------|
|   | 0         | 5       | 10    |
| A | X X X X X |         |       |
| B | - - - -   | X X X   |       |
| C | - -       | - - X X |       |
| D |           | - - - - | X X   |
| E |           | - - - - | - - X |

- Shortest Process Next:

|   |           |         |       |
|---|-----------|---------|-------|
|   | 0         | 5       | 10    |
| A | X X X X X |         |       |
| B | - - - -   | - - - - | X X X |
| C | - -       | X X     |       |
| D |           | - - X X |       |
| E |           | - X     |       |

- Shortest Remaining Time:

|   |           |           |       |
|---|-----------|-----------|-------|
|   | 0         | 5         | 10    |
| A | X - - - - | - - - - X | X X X |
| B | - X X X   |           |       |
| C | - X       | X         |       |
| D |           | - - X X   |       |
| E |           | X         |       |

- Highest Response Ratio Next:

|   |                        |                      |       |
|---|------------------------|----------------------|-------|
|   | 0                      | 5                    | 10    |
| A | X X X X X <sup>1</sup> |                      |       |
| B | - - - -                | X X X <sup>2</sup>   |       |
| C | - -                    | - - X X <sup>3</sup> |       |
| D |                        | - - - -              | - X X |
| E |                        | - - - -              | X     |

zu 1: B:  $\frac{4+3}{3} = 2, \bar{3}$ , C:  $\frac{2+2}{2} = 2$

zu 2: C:  $\frac{5+2}{2} = 3,5$ , D:  $\frac{3+2}{2} = 2,5$ , E:  $\frac{2+1}{1} = 3$

zu 3: D:  $\frac{5+2}{2} = 3,5$ , E:  $\frac{4+1}{1} = 5$

- b) Berechnen Sie die durchschnittliche Abarbeitungszeit und das durchschnittliche Verhältnis zur eigentlichen Bearbeitungszeit. Welcher Algorithmus ist nach diesem Vergleich der bessere? (4 Punkte)

- First Come First Server:

| Prozess      | Bearbeitungszeit (BeZ) | Abarbeitungszeit (Abz) | Verhältnis AbZ/BeZ |
|--------------|------------------------|------------------------|--------------------|
| A            | 5                      | 5                      | 1                  |
| B            | 3                      | 7                      | $2, \bar{3}$       |
| C            | 2                      | 7                      | 3,5                |
| D            | 2                      | 7                      | 3,5                |
| E            | 1                      | 7                      | 7                  |
| Durchschnitt | 2,6                    | 6,6                    | 2,538              |

- Shortest Process Next:

| Prozess      | Bearbeitungszeit (BeZ) | Abarbeitungszeit (Abz) | Verhältnis AbZ/BeZ |
|--------------|------------------------|------------------------|--------------------|
| A            | 5                      | 5                      | 1                  |
| B            | 3                      | 12                     | 4                  |
| C            | 2                      | 4                      | 2                  |
| D            | 2                      | 5                      | 2,5                |
| E            | 1                      | 2                      | 2                  |
| Durchschnitt | 2,6                    | 5,6                    | 2,3                |

- Shortest Remaining Time:

| Prozess      | Bearbeitungszeit (BeZ) | Abarbeitungszeit (Abz) | Verhältnis AbZ/BeZ |
|--------------|------------------------|------------------------|--------------------|
| A            | 5                      | 13                     | 2,6                |
| B            | 3                      | 3                      | 1                  |
| C            | 2                      | 3                      | 1,5                |
| D            | 2                      | 4                      | 2                  |
| E            | 1                      | 1                      | 1                  |
| Durchschnitt | 2,6                    | 4,8                    | 1,846              |

- Highest Response Ratio Next:

| Prozess      | Bearbeitungszeit (BeZ) | Abarbeitungszeit (Abz) | Verhältnis AbZ/BeZ |
|--------------|------------------------|------------------------|--------------------|
| A            | 5                      | 5                      | 1                  |
| B            | 3                      | 7                      | 2,3                |
| C            | 2                      | 7                      | 3,5                |
| D            | 2                      | 8                      | 4                  |
| E            | 1                      | 5                      | 5                  |
| Durchschnitt | 2,6                    | 6,4                    | 2,462              |

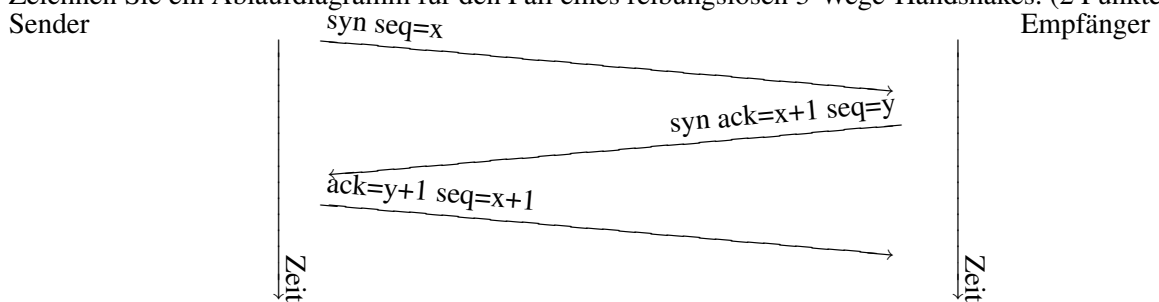
Also Shortest Remaining Time ist nach diesem Vergleich der bessere

- c) Welchen der Scheduling-Algorithmen würden Sie im Kernel für einen typischen Büro-PC einsetzen? Begründen Sie knapp Ihre Entscheidung. (2 Punkte)
- Da aber in typischen Büro-PC in der Regel keine langwierigen Prozesse stattfinden, würde ich einen einfacheres Verfahren nehmen, also z. B. First Come First Server oder Shortest Remaining Time. Vielleicht wäre ein präemptiver Algorithmus wie Shortest Remaining Time auch nicht schlecht, damit Benutzereingaben (z. B. der Mauszeiger) nicht warten müssen.

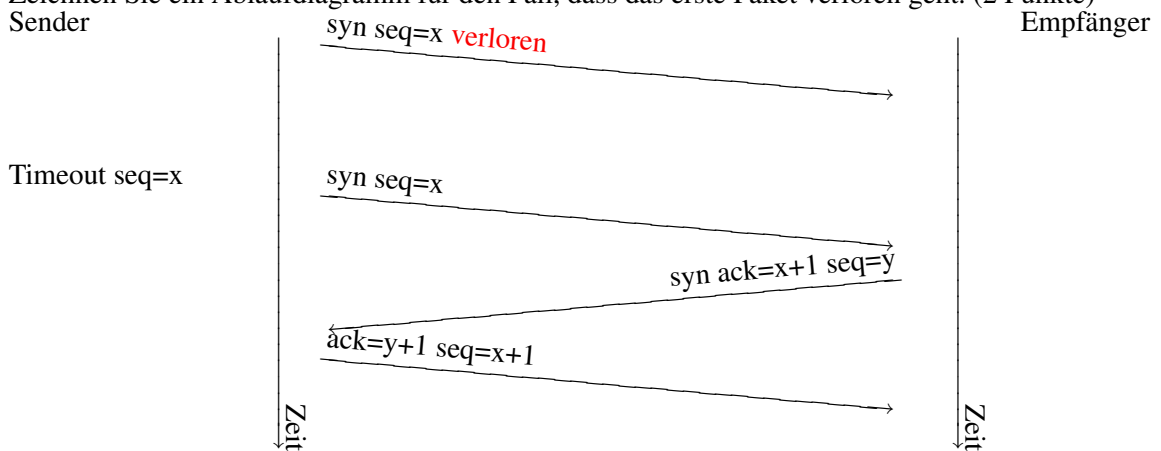
## Aufgabe 6: 3-Wege-Handshake

10 Punkte

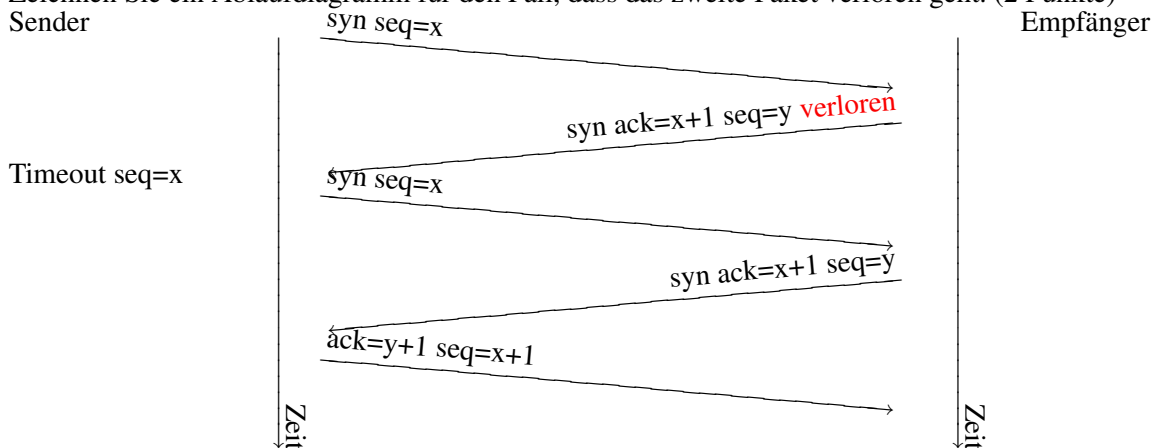
- a) Zeichnen Sie ein Ablaufdiagramm für den Fall eines reibungslosen 3-Wege-Handshakes. (2 Punkte)



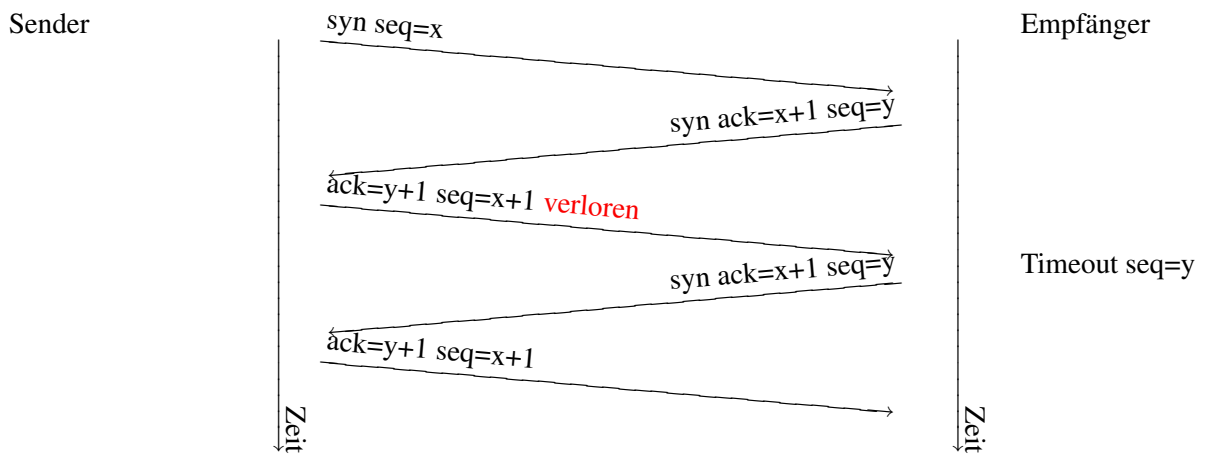
- b) Zeichnen Sie ein Ablaufdiagramm für den Fall, dass das erste Paket verloren geht. (2 Punkte)



- c) Zeichnen Sie ein Ablaufdiagramm für den Fall, dass das zweite Paket verloren geht. (2 Punkte)



- d) Zeichnen Sie ein Ablaufdiagramm für den Fall, dass das dritte Paket verloren geht. (2 Punkte)



- e) Was ist beiden Partnern im Falle eines geglückten 3-Wege-Handshakes bekannt und wozu ist dies nützlich. (2 Punkte)

Nun dieses Verfahren ermöglicht eine sichere Kommunikation über ein sicheres Medium. Nun ist beiden Partnern bekannt, dass sie verbunden sind und auch, dass der jeweils andere Partner, das auch weiß. Der dritte Schritt ist dabei notwendig, damit der Empfänger weiß, ob der Sender die Bestätigung erhalten hat. Das Zweite Paket könnte ja verloren gegangen sein und der Empfänger würde nur alleine wissen, dass die Verbindung zustande gekommen ist, da der Sender keine Bestätigung erhalten hat. Außerdem können so die Teilnehmer die Round Trip Time abschätzen.

## Aufgabe 7: Routing

18 Punkte

- a) Welche Protokolle werden in welcher Reihenfolge aufgerufen, wenn sie mit einem frisch ans LAN angeschlossenen Rechner einen Seitenaufruf in ihrem Browser starten? (4 Punkte)
- Je nach Konfiguration des Rechners wird bei Anschluss des Rechners eine DHCP-Anfrage gestartet, die über  $\text{DHCP} \rightarrow \text{UDP} \rightarrow \text{IP} \rightarrow \text{Ethernet}$  sich eine IP-Adresse, Netzmaske, DNS-Server und ein Gateway holt.
  - Dann wenn die Anfrage gestartet wird, wird zunächst eine DNS-Abfrage an den DNS-Server geschickt, die über  $\text{DNS} \rightarrow \text{TCP/UDP} \rightarrow \text{IP} \rightarrow \text{Ethernet}$  die angefragte Adresse in eine IP-Adresse auflöst.
  - Nun wird eine HTTP Anfrage (wahrscheinlich GET) an diese IP gerichtet, die über  $\text{HTTP} \rightarrow \text{TCP} \rightarrow \text{IP} \rightarrow \text{Ethernet}$  die angefragte Resource holt.
- b) Beschreiben Sie in Stichpunkten die grundsätzliche Funktionsweise eines Link State Routing Algorithmus. Was ist der Hauptunterschied im Gegensatz zu Distance Vector Routing Algorithmen? Welche Vor- und Nachteile bringt dies mit sich? (8 Punkte)

**Link State Routing:** • Router sendet Pakete zur Entdeckung der Nachbarn

- Router misst zu allen **Nachbarn** die Entfernungen (Hörs oder Verzögerung)
- Die diese Informationen werden an alle anderen Router verteilt (Fluten mit Erweiterungen zur Vermeidung von Duplikaten und Aging).
- Bestimmung der kürzesten Wege (z.B. mit Dijkstra) an Hand der verteilten Daten

**Distance Vector Routing:** • Der Router erzeugt eine Routingtabelle mit **allen** anderen Routern im Netz und der geschätzten Entfernung zu diesen und dem entsprechenden nächsten Hop.

- Informationen über entfernte Router erhält er über seine 1-Hop Nachbarn.
- DVR muss Count-to-Infinity Problem lösen.

**Vorteile:** • Schneller

- Skaliert besser.
- Kein Count-to-Infinity Problem

**Nachteile:** • Wesentlich kompliziert und nicht so leicht zu warten wie Distance Vector Routing

- c) Was ist das Problem mit Klasse A- und B-Netzwerken und was tut man dagegen? (3 Punkte)

Klasse A- und Klasse B-Netzwerke umfassen sehr viele IP-Adressen, die meistens von den Institutionen, denen diese Adressen zugewiesen sind, gar nicht voll ausgeschöpft werden. Ein Klasse A Netzwerk umfasst zum Beispiel  $2^{24}$  (etwa 16,8 Mio.) mögliche IP Adressen, Klasse B Netzwerke umfassen  $2^{16}$  Adressen. In Folge des stark wachsenden Bedarfs an IP Adressen hat man sowohl CIDR

(Classless Inter Domain Routing) als auch NAT (Network Address Translation) eingeführt. Zur Zeit wird an der Einführung von IPv6 gearbeitet.

- d) Nennen jeweils eine Anwendungen, bei der Broadcast-, Multicast- bzw. Anycast-Pakete zum Einsatz kommt. (3 Punkte)

**Broadcast:** DHCP

**Multicast:** IPTV

**Anycast:** DNS Root Nameserver

## Aufgabe 8: Datenrate

5 Punkte

Ein Signal wird über in 6 Signalstufen über einen 4 kHz-Kanal gesendet

- a) Berechnen Sie die maximale Datenrate, wenn der Kanal rauschfrei ist. (2 Punkte)  
Wende die Formel von Nyquist an:

$$\begin{aligned}\text{Maximale Datenrate} &= 2H \log_2 V \text{ bit/s} \\ &= 2 \cdot 4000 \cdot \log_2 6 \text{ bit/s} \\ &\approx 8000 \cdot 2,6 \text{ bit/s} \\ &= 20.800 \text{ bit/s}\end{aligned}$$

- b) Berechnen Sie die maximale Datenrate, wenn das Signal-zu-Rausch-Verhältnis des Kanals 30dB beträgt. Die Umrechnung von x dB zur S/N lautet  $x=10 \cdot \log_{10}(S/N)$ . (3 Punkte)  
Rechne erst mal die die 30dB in  $\frac{S}{N}$  um.

$$\begin{aligned}30 \text{ dB} &= 10 \cdot \log_{10} \left( \frac{S}{N} \right) \\ 3 \text{ dB} &= \log_{10} \left( \frac{S}{N} \right) \\ \frac{S}{N} &= 10^3 = 1000\end{aligned}$$

Wende nun die Formel von Shannon an:

$$\begin{aligned}\text{Maximale Datenrate} &= H \log_2 \left( 1 + \frac{S}{N} \right) \text{ bit/s} \\ &= 4000 \cdot \log_2 (1001) \text{ bit/s} \\ &\approx 4000 \cdot 10 \text{ bit/s} \\ &= 40.000 \text{ bit/s}\end{aligned}$$