Exercise Sheet 3

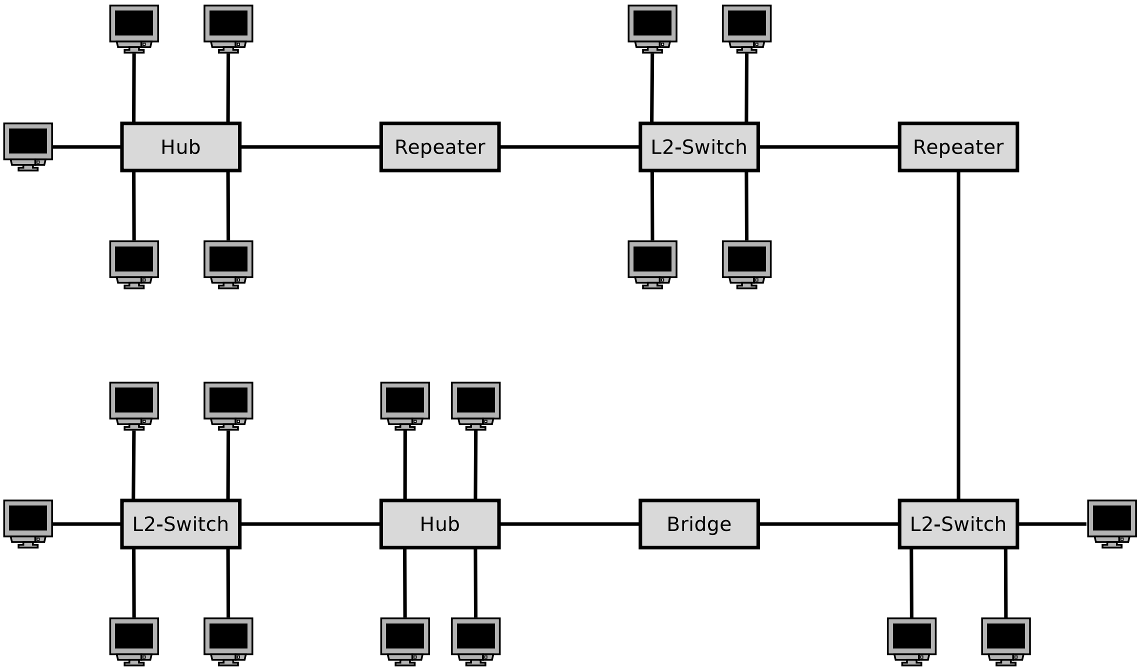
Exercise 1 (Bridges and Switches)

1. Explain what the purpose of **Bridges** in computer networks is.
2. Give the number of **interfaces** („*Ports*") a Bridge provides.
3. Explain what the major difference between **Bridges** and **Layer-2-Switches** is.
4. Explain why Bridges and Layer-2-Switches do not require **physical or logical addresses**.
5. Name at least two **examples** of Bridge implementations.
6. Explain what the advantage of **Learning Bridges** is in contrast to „dumb" Bridges.
7. Name the information that is stored in the **forwarding tables** of Bridges.
8. Explain what happens, if for a network device, no entry exists in the **forwarding table** of a Bridge.
9. Explain why Bridges try to avoid **loops**.
10. Name the protocol that Bridges implement for **handling loops**.
11. Explain what a spanning tree is.
12. Give the information, the **Bridge ID** according to the IEEE contains.
13. Explain what the difference between the **Bridge ID** according to the IEEE and the **Cisco extended version** of the Bridge ID is.
14. Give the number of priority values that can be encoded with the **Bridge ID** according to the IEEE.
15. Give the number of priority values that can be encoded with the **Cisco extended version** of the Bridge ID.
16. Explain what a **Bridge Protocol Data Unit** (BPDU) message is and for what purpose it is used.
17. Give the selection criteria for determining, whether a Bridge becomes the **Root Bridge**.
18. Explain what a **Designated Bridge** is and what its task is.
19. Give the number of **Designated Bridges**, a computer network contain.

20. Give the selection criteria for determining, whether a Bridge becomes a **Designated Bridge**.
21. Explain what the impact of Bridges and Layer-2-Switches on the **collision domain** is.
22. Explain what a switched network is.
23. Explain the benefit of a switched network compared to a non-switched network.

Exercise 2 (Collision Domain)

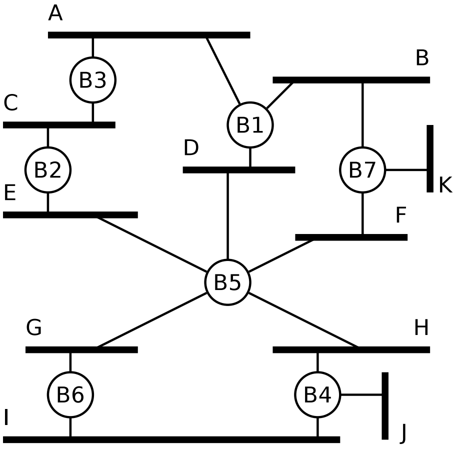
Sketch in the diagram all **collision domains**.



Exercise 3 (Spanning Tree Protocol)

The figure shows the physical connections of a network topology. A-J are physical networks (LANs). B1-B7 are Bridges (L2-Switches). All Bridges boot up at the same time after a power failure. Highlight in the figure which ports and Bridges are not used when the Spanning Tree Protocol is used.

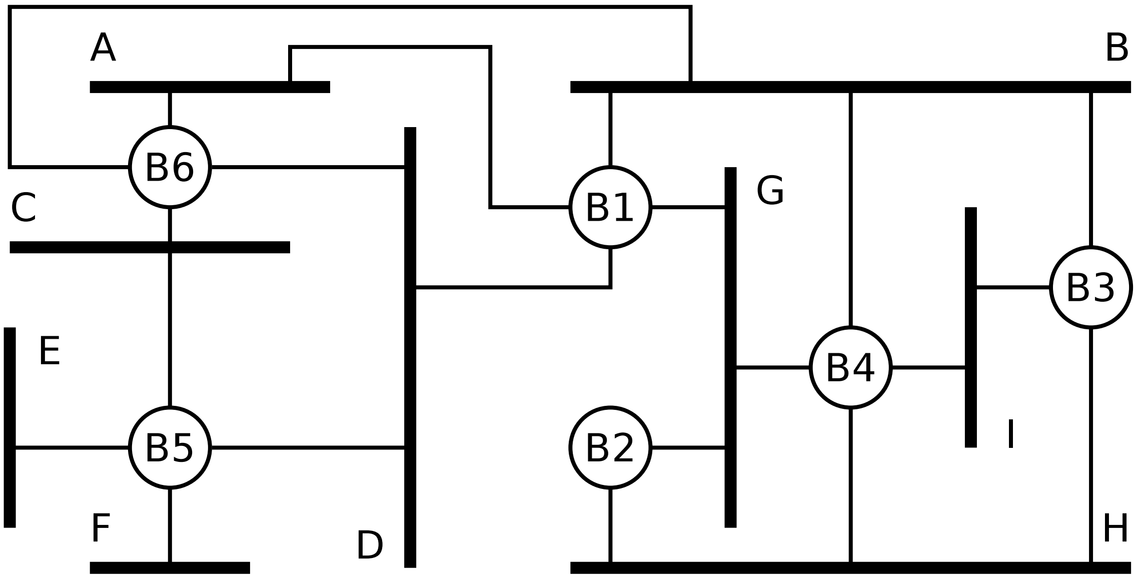
Attention: If multiple paths from a network to the root bridge have the same distance, then take the bridge IDs as decision criterion. The smaller the ID of a bridge is, the higher is its priority.



Exercise 4 (Spanning Tree Protocol)

The figure shows the physical connections of a network topology. A-I are physical networks (LANs). B1-B6 are Bridges (L2-Switches). All Bridges boot up at the same time after a power failure. Highlight in the figure which ports and Bridges are not used when the Spanning Tree Protocol is used.

Attention: If multiple paths from a network to the root bridge have the same distance, then take the bridge IDs as decision criterion. The smaller the ID of a bridge is, the higher is its priority.



Exercise 5 (Addressing in the Data Link Layer)

1. Data Link Layer protocols specify the **format** of…

   □ physical network addresses □ logical network addresses

2. Give the name (technical term) of **physical network addresses** (Data Link Layer addresses).

3. Name the protocol that is used by Ethernet for the **address resolution**.

4. Which devices receive a frame with the **destination address** `FF-FF-FF-FF-FF-FF`.

5. Explain what **MAC spoofing** is.

Exercise 6 (Framing)

1. One way to mark the frames' borders is via **character count in the frame header**. Name a potential issue that can arise from this method.

2. One way to mark the frames' borders is via **Byte Stuffing**. Name a drawback of this method.

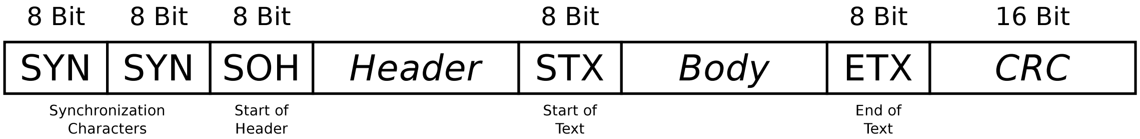
3. Explain why up-to-date Data Link Layer protocols, such as Ethernet and WLAN, work **bit-oriented and not byte-oriented**.

4. Mark the information that an **Ethernet frame** contains.

   □ Sender IP address
   □ Sender MAC address
   □ Hostname of the receiver
   □ Information about the Transport Layer protocol used
   □ Preamble to synchronize the receiver
   □ Port number of the receiver
   □ CRC checksum
   □ Information about the Application Layer protocol used
   □ VLAN tag
   □ Receiver MAC address
   □ Receiver IP address
   □ Information about the Network Layer protocol used
   □ Hostname of the sender
   □ Signals, which are transmitted via the transmission medium
   □ Port number of the sender

Exercise 7 (Byte Stuffing)

The Data Link Layer splits the bit stream from the Physical Layer into frames. The character-oriented protocol BISYNC uses control characters to mark the structure of the frames. The start of a frame highlights the character SYN. The start of the header highlights the character SOH (*Start of header*). The payload is located between STX (*Start of text*) and ETX (*End of text*). The figure shows the structure of BISYNC frames:



Control character	SOH	STX	ETX	DLE	SYN
Hexadecimal notation	01	02	03	10	16

If the payload (body) contains the control characters ETX and DLE (*Data Link Escape*), they are protected (*escaped*) by the Data Link Layer protocol with a stuffed DLE caracter. A single ETX in the payload area is represented by the sequence DLE ETX. The DLE character itsef is represented by the sequence DLE DLE.

Mark the payload inside the following BISYNC frames.

1. 16 16 01 99 98 97 96 95 02 A1 A2 A3 A4 A5 03 A0 B7
2. 16 16 01 99 98 97 96 95 02 05 04 10 03 02 01 03 76 35
3. 16 16 01 99 98 97 96 95 02 10 03 10 10 10 03 03 92 55
4. 16 16 01 99 98 97 96 95 02 10 10 10 10 10 03 01 02 A1 03 99 B2

Source: Jörg Roth. Prüfungstrainer Rechnernetze. Vieweg (2010) and Wikipedia

Exercise 8 (Bit Stuffing)

The Data Link Layer protocol HDLC (High-Level Data Link Control) uses Bit Stuffing. If the sender discovers 5 consecutive 1 bits in the bitstream from the Network Layer, it *stuffs* a single 0 bit into the outgoing bit stream. If the receiver discovers 5 consecutive 1 bits, followed by a single 0 bit in the bit stream from the Physical Layer, it removes (*destuffs*) the 0 bit.

Give the encoding for each one of the following bit sequences, when the sender *stuffs* after 5 consecutive 1 bits a single 0 bit into the bit stream from the Network Layer.

1. 01111110 10100111 11111000 11110010 10011111 10111111 11100101

2. `00111111 01110001 11110011 11111100 10101010 11001111 11100001`

3. `11111111 11111111 11111111 11111111 11111111 11111111 11111111`

Exercise 9 (Error Detection – CRC)

1. Calculate the frame to be transferred.

   ```
   Generator polynomial: 100101
   Payload: 11010011
   ```

2. Check, if the received frame was transmitted correctly.

   ```
   Transferred frame: 1101001110100
   Generator polynomial: 100101
   ```

3. Check, if the received frame was transmitted correctly.

   ```
   Transferred frame: 1101001111100
   Generator polynomial: 100101
   ```

4. Calculate the frame to be transferred.

   ```
   Generator polynomial: 100101
   Payload: 10110101
   ```

5. Check, if the received frame was transmitted correctly.

   ```
   Transferred frame: 1011010110110
   Generator polynomial: 100101
   ```

6. Check, if the received frame was transmitted correctly.

   ```
   Transferred frame: 1011010110100
   Generator polynomial: 100101
   ```

7. Check, if the received frame was transmitted correctly.

   ```
   Transferred frame: 1010010110100
   Generator polynomial: 100101
   ```

8. Calculate the frame to be transferred.

   ```
   Generator polynomial: 100000111
   Payload: 1101010101110101
   ```

9. Check, if the received frame was transmitted correctly.

   ```
   Transferred frame: 110101010111110110110111
   Generator polynomial: 100000111
   ```

10. Check, if the received frame was transmitted correctly.

```
Transferred frame: 110101010111010110110111
Generator polynomial: 100000111
```

Exercise 10 (Error Correction – Simplified Hamming Code)

Transmission errors can be detected via CRC checksums. If it is important to not only recognize errors, but also to be correct them, then the data to be transmitted must be encoded in a way, that error-correction is possible. Error correction can be realized e.g. via the **Simplified Hamming Code** we discussed in the computer networks course.

1. A message of 8 bits payload (`10011010`) needs to be transferred. Calculate the message, that will be transmitted (payload inclusive parity bits).
2. The following messages have been received. Verify, if they were transmitted correctly.

   a) `00111101`

   b) `101110100010`

   c) `001101100100`

   d) `0001101100101101`

Exercise 11 (Media Access Control)

1. Explain why computer networks use protocols for **media access control**.
2. Explain why Ethernet and WLAN implement different **media access control methods**.
3. Explain how Ethernet devices react, when they detect a **collision**.
4. Explain why it is important that the transmission of a frame is not completed when a collision occurs in an Ethernet network.
5. Explain what is done to ensure that the transmission of a frame is not completed when a collision occurs in an **Ethernet** network.
6. Name the two special characteristics of the transmission medium in wireless networks that cause undetected collisions at the receiver.

7. Describe both special characteristics of subtask 6.
8. Explain what the Network Allocation Vector (NAV) is and for what purpose it is used.
9. Explain what the Contention Window (CW) is and for what purpose it is used.
10. Name a benefit and a drawback of using the control frames **Request To Send** (RTS) and **Clear To Send** (CTS).

Exercise 12 (Address Resolution Protocol)

1. Explain what the function of the **Address Resolution Protocol** is.
2. Explain what the **ARP cache** is.