

1. Virtual LANs

- (a) **VLAN-Bridges** Einer der Hauptvorteile von Bridges ist die autoconfiguration der Forwarding Tabellen. Funktioniert dies auch noch, wenn die Bridge VLANs weiterleiten soll? (Nehmen Sie an, dass alle Geräte im LAN IEEE 802.1q implementieren.)

lar – separate Behandlung je nach VLAN Tag. Ports werden zusätzlich automatisch nach VLAN-Tags eingefärbt. Details sind ähnlich den spanning tree Verfahren.

- (b) **Zustand in VLAN-Bridges?**

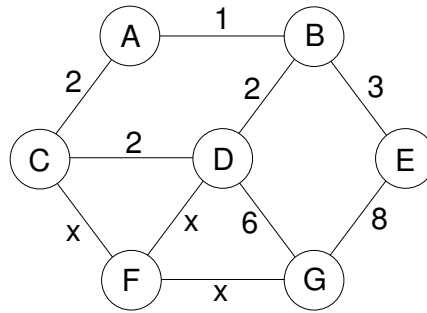
In welchem Sinne wird durch IEEE 802.1q VLAN Zustand in Bridges bzw. Switches erzeugt? Ist dies eine Verletzung des Prinzips eines zustandslosen Netzes?

a, die VLAN-Tabellen kann man als Zustand auffassen, und die VLAN-Ids im Packet als Connection Identifier.

Andererseits kann man argumentieren, dass dies lediglich eine erweiterte Zieladresse ist. Und es sich ja ohnehin bestenfalls um soft state handelt, der automatisch erzeugt wird. Es also keinen Verbindungsaufbau bzw. Zustandserzeugung und -beseitigung bedarf.

2. Dijkstra-Algorithmus

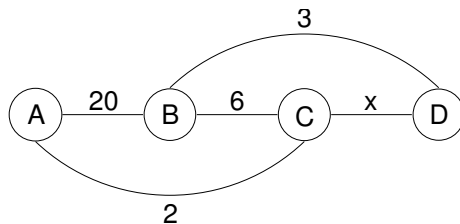
Gegeben sei folgende Netzwerktopologie:



- Sei $x = 3$. Bestimmen Sie den kürzesten Weg von A nach G mit Hilfe des Dijkstra-Algorithmus. Geben Sie für jeden Iterationsschritt den Arbeitsknoten und die Knotenmarkierungen an.
- Geben Sie einen Wert für x an, der den Ausfall des Knotens F modelliert.
- Bestimmen Sie jetzt den kürzesten Weg von A nach G mit Hilfe des Dijkstra-Algorithmus unter der Annahme, dass Knoten F tatsächlich ausgefallen ist. Geben Sie für jeden Iterationsschritt den Arbeitsknoten und die Knotenmarkierungen an.

3. Distanzvektor-Wegwahlverfahren (Bellman-Ford Routing)

Gegeben sei folgende Netztopologie:



Die folgende Tabelle zeigt spaltenweise die initialen Distanzvektoren:

	A	B	C	D
A	$0 \rightarrow A$	-	-	-
B	-	$0 \rightarrow B$	-	-
C	-	-	$0 \rightarrow C$	-
D	-	-	-	$0 \rightarrow D$

- Führen Sie das Distanz-Vektorverfahren für $x = 10$ aus. Tragen Sie die endgültigen Distanzvektoren in die Tabelle ein. Notieren Sie sowohl die Kosten als auch den Nachbarn, über den der jeweils beste bekannte Weg verläuft.
- Die Verbindung x verbessert sich zu $x = 1$. Führen Sie das Distance Vector-Verfahren erneut aus, basierend auf den Distanzvektoren aus Ihrer Lösung zu Teil 3a.

4. Count-to-Infinity Problem

Gegeben sei ein Netz der Form $A-B-C-D-E$, bei dem alle Kanten die Kosten 1 haben. Nehmen Sie an, der Knoten E sei zu Beginn abgeschaltet und alle anderen Knoten haben korrekte Routing-Tabellen.

- (a) Knoten E wird eingeschaltet. Zeigen Sie, wie die Routing-Tabellen in den anderen Knoten auf diese Änderung hin durch den Distance Vector Algorithmus geändert werden.
- (b) Was geschieht, wenn Knoten E wieder abgeschaltet wird (nehmen Sie an, dass zuvor der Algorithmus konvergiert ist.)? Wie reagiert der Algorithmus auf diese Änderung? Vergleichen Sie die Konvergenzgeschwindigkeit in beiden Fällen und erklären Sie, warum man hier vom *Count-to-Infinity* Problem spricht.

5. Bearbeiten Sie das Notebook zu Bellmann-Ford Routing.