12. Übung zur Vorlesung
"Vernetzte Systeme"
WS 2000/2001

Prof. Dr. F. Mattern

Ausgabedatum: 24. Jan. 2001
Abgabedatum: 31. Jan. 2001

Aufgabe 42 (Shortest Path Routing)

Viele Routingalgorithmen basieren auf sogenannten *Shortest Path Algorithmen*, welche den "billigsten" (bzw. "kürzesten") Weg zwischen einem Anfangs- und einem Zielknoten berechnen. Einer davon wurde von Dijkstra vorgeschlagen und funktioniert folgendermassen:

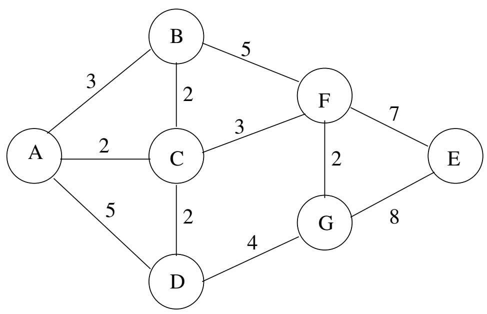
Jeder Knoten wird mit seinem kürzesten bekannten Abstand[1] zum Quellknoten und dem Namen des Vorgängerknotens auf diesem kürzesten Weg beschriftet. Ist kein Weg bekannt, werden die Knoten mit ∞ beschriftet. Die Beschriftungen können vorläufig (weiter veränderbar) oder permanent (endgültig) sein. Man beginnt, indem man den Anfangsknoten zum aktuellen Arbeitsknoten erklärt, als permanent markiert und alle benachbarten Knoten vorläufig mit ihrem Abstand zu diesem Arbeitsknoten beschriftet.

Danach untersucht man alle vorläufig markierten Knoten im ganzen Graphen, markiert den mit kleinstem Abstand als permanent und macht ihn zum neuen Arbeitsknoten.

Während der Algorithmus arbeitet und neue Wege gefunden werden, können sich die vorläufigen Beschriftungen zugunsten eines kürzeren Weges ändern.

Der Algorithmus terminiert, wenn der Zielknoten als permanent markiert wird. Der kürzeste Weg kann nun den Markierungen nach rekonstruiert werden.[2]

(6 Punkte) Bestimmen Sie für folgendes Netz den optimalen Weg von A nach E gemäss dem Shortes Path Algorithmus von Dijkstra. Geben Sie dabei alle Schritte bei der Berechnung an. Benutzen Sie dafür die Schablone auf der letzten Seite dieses Übungsblattes.



[1] Der Abstand zweier Knoten bezüglich eines Pfades ist die Summe der Kosten der Kanten auf diesem Pfad.
[2] Für weitere Informationen zum Algorithmus siehe z.B. A. Tanenbaum "Computer Networks".

Aufgabe 43 (Paketverschachtelung)

Wie aus der Vorlesung bereits bekannt ist, wird ein Datenpaket auf jedem Layer um einen Header ergänzt. Die Strukturen der wichtigsten Header sind hier abgebildet:

Für Ethernet-Pakete:

bytes	7	1	6	6	2
	Preamble	SFD	DA	SA	Length

Legende: SFD (Start-frame delimiter),
DA (Destination address),
SA (Source address)

Für IP-Pakete:

bits 0 4 8 16 19 31

Version	HL	Type of Service	Total Length	
Identification			Flags	Fragment Offset
Time to Live		Protocol	Header Checksum	
Source Address				
Destination Address				

Legende: HL (Header length)

Für TCP-Pakete:

bits 0 4 10 16 31

Source Port								Destination Port
Sequence Number								
Acknowledgement Number								
DO	Reserved	U	A	P	R	S	F	Window Size
Checksum								Urgent Pointer

Legende: DO (Data offset),
U (URG -Urgent pointer),
A (ACK - Acknowledgment),
P (PSH - Push function),
R (RST - Reset the connection),
S (SYN - Syncronize the sequence numbers),
F (FIN - No more data from sender)

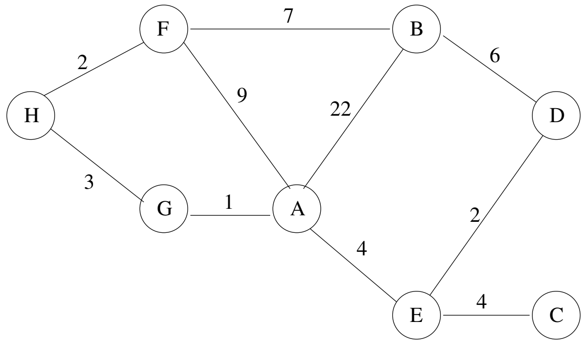
(6 Punkte) Folgender Bit-String stellt ein Ethernet-Paket dar, das ein IP-Paket enthält, welches ein TCP-Paket enthält. Extrahieren Sie die Informationen aus diesen Headern. Geben Sie Ethernet-Adressen im Hexadezimalformat an, Internet-Adressen in der viergliedrigen Dezimalform, Längenangaben , Port-Nummern etc. als Dezimalzahlen.

01010101 01010101 01010101 01010101 01010101 01010101 01010101 10101011
00001000 00000000 00100000 10101001 10000011 01000100 00000000 01010000
00000100 10101100 10000100 11111010 00000000 00101000 01000101 00000000
00000000 00101000 01010011 10110010 01000000 00000000 11111111 00000110
00001010 10001100 10000001 10000100 00001101 01000010 10000001 10000100
00001101 01000111 10100111 10101011 00000000 00010111 11000001 11011111

10001010 00011010 01110111 00011001 11110100 11011000 01010000 00010000
00100010 00111000 00010000 01011100 00000000 00000000

Aufgabe 44 (Source Routing)

(4 Punkte) Im folgenden wird das Beispielnetz aus Aufgabe 41 wiederverwendet. Die Knoten symbolisieren hier Brücken. Knoten C sendet einen *discovery frame* an Knoten H. Geben Sie an, welche Pakete C schliesslich zurückerhält und wie dort jeweils die RD-Felder gesetzt sind.[3]



[3] Dieser *Route-Discovery*-Algorithmus wurde in der Vorlesung übersprungen, ist aber in Ihrer Foliensammlung zu finden.

Anhang: Schablone für die Lösung von Aufgabe 42