Opgaveregning 10 (14-04-2020)

## Chapter 4

### Review 20:

**When a large datagram is fragmented into multiple smaller datagrams, where are these smaller datagrams reassembled into a single larger datagram?**

De bliver samlet ved deres destination, og de behøver ikke at tage den samme vej derhen.

### Review 25:

**Suppose an application generates chunks of 40 bytes of data every 20 msec, and each chunk gets encapsulated in a TCP segment and then an IP datagram. What percentage of each datagram will be overhead, and what percentage will be application data?**

Payload = 40 bytes  
TCP header = 20 bytes  
IP header = 20 bytes  
(40 + 20 + 20) / (20 + 20) = 50% er headers

### Problem 8:

**Consider a router that interconnects three subnets: Subnet 1, Subnet 2, and Subnet 3. Suppose all of the interfaces in each of these three subnets are required to have the prefix 223.1.17/24. Also suppose that Subnet 1 is required to support up to 62 interfaces, Subnet 2 is to support up to 106 interfaces, and Subnet 3 is to support up to 15 interfaces. Provide three network addresses (of the form a.b.c.d/x) that satisfy these constraints.**

Subnet 1: Sidste byte: 10xxxxxx. IP: 223.1.17.128/26

Subnet 2: Sidste byte: 0xxxxxxx. IP: 223.1.17.0/25

Subnet 3: Sidste byte: 110xxxxx. IP: 223.1.17.192/27

### Problem 11:

**Consider a subnet with prefix 192.168.56.128/26. Give an example of one IP address (of form xxx.xxx.xxx.xxx) that can be assigned to this network.**

192.168.56.129

**Suppose an ISP owns the block of addresses of the form 192.168.56.32/26. Suppose it wants to create four subnets from this block, with each block having the same number of IP addresses. What are the prefixes (of form a.b.c.d/x) for the four subnets?**

Fra solutions:  
Subnet 192.168.56.32/26. The exercise is wrong. In fact, the IP address should have 26 bits for the network part., and the rest should be 0 (it is the address of the network). Anyway, 32 = 0010 0000.

Let us solve a similar exercise with Subnet 192.168.56.64/26. It means that all the addresses owned by the ISP have the last byte of the form:

01xx xxxx

The ISP will create 4 subnets by using the first two free bits, which will not be host part of the address anymore:

0100 xxxx

0101 xxxx

0110 xxxx

0111 xxxx

Thus the 4 subnets have network address:

192.168.56.64/28

192.168.56.64/80

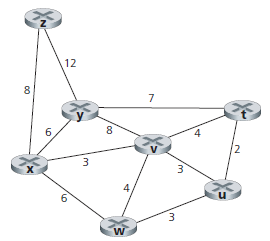
192.168.56.64/96

192.168.56.64/112

## Chapter 5

### Problem 3:

**Consider the following network. With the indicated link costs, use Dijkstra’s shortest-path algorithm to compute the shortest path from *x* to all network nodes. Show how the algorithm works by computing a table similar to Table 5.1.**

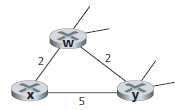


Tabel:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Step | N´ | D(t) | D(u) | D(v) | D(w) | D(y) | D(z) |
| 0 | x | NaM | NaM | 3,x | 6,x | 6,x | 8,x |
| 1 | xv | 7,v | 6,v |  | 6,x | 6,x | 8,x |
| 2 | xvu | 7,v |  |  | 6,x | 6,x | 8,x |
| 3 | xvuw | 7,v |  |  |  | 6,x | 8,x |
| 4 | xvuwy | 7,v |  |  |  |  | 8,x |
| 5 | xvuwyt |  |  |  |  |  | 8.x |
| 6 | xvuwytz |  |  |  |  |  |  |

### Problem 7:

**Consider the network fragment shown below. *x* has only two attached neighbors, *w* and *y*. *w* has a minimum-cost path to destination *u* (not shown) of 5, and *y* has a minimum-cost path to *u* of 6. The complete paths from *w* and *y* to *u* (and between *w* and *y*) are not shown. All link costs in the network have strictly positive integer values.**



1. **Give *x*’s distance vector for destinations *w, y,* and *u*.**

Dx(w)=2

Dx(y)=4 (først w så y)

Dx(u)=7 (først w så u)

1. **Give a link-cost change for either *c*(*x,w*) or *c*(*x,y*) such that *x* will inform its neighbors of a new minimum-cost path to *u* as a result of executing the distance-vector algorithm.**

Lige pt er ruten af x til u: (x,w,u) hvis vi gerne i stedet til have den til at se sådan her ud: (x,y,u) så kan vi ændre værdien mellem x og w til at være højere sådan at ruten mellem x og y nu i stedet er det path som koster mindst, hvilket vil betyde at x vil informere sine naboer omkring denne ændring.

1. **Give a link-cost change for either *c*(*x,w*) or *c*(*x,y*) such that *x* will *not* inform its neighbors of a new minimum-cost path to *u* as a result ofexecuting the distance-vector algorithm.**

Hvis man fx gør værdien mellem x og w 1 større så ændre det ikke noget i det store billede, da det stadig er den bedste path at tage.

## Chapter 6

### Review 9:

**How big is the MAC address space? The IPv4 address space? The IPv6 address space?**

MAC: 2^48 = 281.474.976.710.656

IPv4: 2^32 = 4.294.967.296

IPv6: 2^128 = 340.282.366.920.938.463.463.374.607.431.768.211.456