Dept. Math. & Comp. Sc.	**Computer Networks**
Vrije Universiteit	**10.06.2002**

Always explain your answers concisely and be sure to be to-the-point.

Part I

This part covers the same material as the midterm exam.

1a What is the role of the session layer in the OSI reference model? Give two typical examples of services that belong in this layer. *5pt*

This layer is used to support a series of communication actions (i.e, a session) that cannot be captured by the services offered by the transport layer. Typical examples include user dialogue management and continuation of long-file transfer after the original connection was broken.

1b The presentation layer transforms application-level data into a machine- and network-independent format. Is this transformation also necessary when transferring files? *5pt*

No, the conversion (marshalling) is needed only when sending messages. When sending data contained in files, the sending and receiving application have already reached agreement on the exact format of the data.

1c One could argue that the application layer should be split between a sublayer that contains general-purpose protocols and one that contains application-specific protocols, as shown below. Give an example of each type of protocol. *5pt*

Application-specific protocols
General-purpose application protocols
Presention layer
...

A banking protocol as used for automatic teller machines, is an example of an application-specific protocol. A protocol for handling faults, as well as many security protocols are examples of general-purpose application protocols.

2a What is the difference between an analog and a digital signal? *5pt*

An analog signal is continuous; a digital signal is discrete. More precisely, for each instant of time t *and arbitrary small value of* $\delta > 0$*, an analog signal will show a smooth transition to the value at* $t + \delta$*. A digital signal is defined on regular intervals* $[k \cdot t, k \cdot t + \Delta]$*,* $k \geq 0$ *and* Δ *fixed. For each interval, the signal value is constant and from a finite number of possible values.*

2b Consider an audio signal that is sampled at a frequency of f Hz with a sample size of b bits.

(b1) What is the required bandwidth to send this signal uncompressed across a network? *1pt*

(b2) Considering that humans cannot hear audio tones higher than 20 kHz, what is a reasonable maximum value for f? *1pt*

(b3) What is the effect of taking $b = 1$? And $b = 32$? *3pt*

(b1) $f \times b$ *bits/s. (b2) Nyquist tells us that* $f > 40$ *kHz is useless. (b3) Garbage for* $b = 1$*: there are now only two signal values by which it is impossible to decode the received signal to a reasonable audio signal. A very high value of* b *allows to accurately encode the various values of the analog source.*

3a The data link layer converts raw bit streams into frames. Why is this conversion necessary? *5pt*

Only by using separate and distinguishable frames it becomes possible to add redundant bits that allow for error detection and correction.

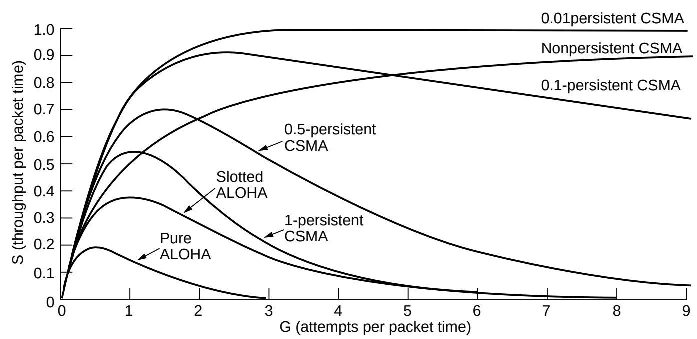
3b Consider a two-layered protocol stack in which layer L_{low} provides an interface for unreliably sending and receiving frames at B_{low} bits per second. Layer L_{high} offers exactly the same interface, but with higher transmission reliability. How can this higher reliability be achieved without making use of retransmissions, and what effect does this have on B_{high}? *5pt*

Layer L_{high} should add more bits to its frame that allow the receiving side to apply error correction *techniques. As a consequence, $B_{high} < B_{low}$.*

4a Explain why the use of slots in slotted Aloha doubles the maximum throughput in comparison to pure Aloha. *5pt*

Using slots prevents a sender to submit a frame at an arbitrary moment, thus avoiding frame collision during the transmission of another frame. At worst, several senders now start sending simultaneously, but there is no need to take into account that your frame is damaged because someone interferred somewhere halfway your transmission.

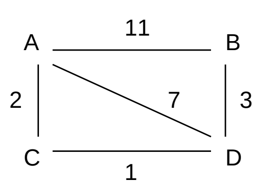
4b Can we conclude from the following figure that *p*-persistent CSMA is always better than *q*-persistent CSMA for $p < q$? Explain your answer. *5pt*



No, they only thing we can say is that of the frames that are sent, a higher percentage will make it safely, i.e. without colliding, to the end.

Part II

5a Consider the following network. Using distance vector routing, what is the distance to *B* that *A* will eventually store in its tables? Explain your answer! *5pt*



A will initially record a distance 11, but as soon as D announces a distance 3 to B, A will update its route via D with distance 10. C will eventually announce a route (through D) of distance 4, which will lead to an update at A of distance 6 (through C).

5b Suppose that the link *BD* breaks. What happens then? *5pt*

All nodes will look for a new path to B. The actual series of updates may differ, but eventually all nodes will discover a path to B through A. D will route through C.

5c Suppose that after some time, the link *AB* also breaks. What happens then? *5pt*

We'll now bump into the count-to-infinity problem. For example, A will search for a new path to B and discover that C knows how to get there (at distance 12). A will update its tables and announce a new distance of 15, eventually leading to a new update at C, and so on.

6a Which problem does the three-way handshake protocol solve? *5pt*

Without a three-way handshake it is possible to start a connection set-up based on an incorrect (old) sequence number from the sender. This sequence number is derived from an old packet that arrived at the receiver after a crash and subsequent restart of the original sender.

6b Is it possible to release a connection such that both parties *always* agree? Explain your answer. *5pt*

No. The problem is that the last message can always be lost so that one of the two parties can only hope that the other knows there is agreement, or even that the connection is going to be torn down.

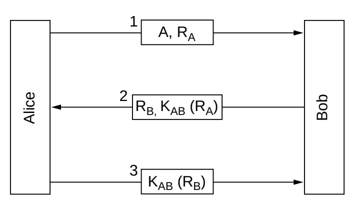
6c Assume the transport layer has a limited number of buffers available for managing reliable connections. How can this lead to a deadlock situation between sender and receiver, and how is this problem solved? *5pt*

The reeciver will need to send buffer credit grants to the sender, telling the sender how much data it can send. When the sender had no more credit, but the new credit message is lost, we wil find ourselves in a deadlock. This deadlock is solved by letting the receiver set a timer and retransmitting the credit message when it has not received any data packets from the sender before the timer goes off.

6d Explain how inappropriate buffer management in TCP can lead to the *silly window syndrome*. *5pt*

This problem occurs when a single byte in the otherwise full receiver buffer becomes available. The receiver may decide to tell the sender that it can send another byte, after which the buffer becomes full again. In effect, only a single byte at time is transferred. The problem can be avoided by giving only credits for a reasonably large number of bytes, and also let the sender stick to always sending a minimum number of bytes.

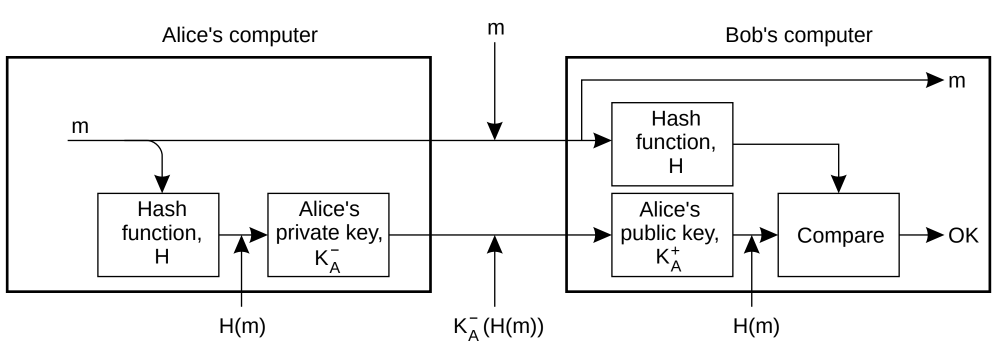
7a What is a fundamentally weak point in the following authentication protocol? *5pt*



It can be subject to a reflection attack by which Trudy can pretend she is Alice. See Fig. 7-14.

7b Provide a simple, efficient protocol so that Alice can digitally sign a public (i.e., nonconfidential) document and send it securely to Bob. *5pt*

Construct a message digest using a hash function, and sign that digest using Alice's private key:



Final grade: *(1) Add, per part, the total points. (2) Let T denote the total points for the midterm exam ($0 \leq T \leq 45$); D1 the total points for part I; D2 the total points for part II. The final number of points E is equal to* $\max\{T, D1\} + D2 + 10$.