CP 372, Winter 2014

Assignment 2 (15%)   **Due Monday, March 10, 19:59 p.m.**

In this assignment you will implement two reliable transfer protocols on top of DatagramSocket API provided by Java:

1) **Stop-and-wait 3.0** as described in the text and
2) **Go-Back-N** with variable window size.

Both implementations split into two parts running on different end systems: **Sender (Client)** and **Receiver (Server)**.

Sender program has 5 (or 6 for Go-Back-N sender) command line arguments:

- <host address of the receiver>
- <UDP port number used by the receiver to receive data from the sender>
- <UDP port number used by the sender to receive ACKs from the receiver>
- <name of the file to be transferred>
- <reliability number>
- <Go-Back-N window size> (which is less or equal to 128)

in the given order.

Receiver program has 4 command line arguments:

- <host address of the sender>
- <UDP port number used by the sender to receive ACKs from the receiver>
- <UDP port number used by the receiver to receive data from the sender>
- <name of the file to write received data>

in the given order.

Sender will open the file and send its content in the series of datagrams, each containing piece of file data (no more than 124 bytes) and sequence number. Sender will also send special EOT (End Of Transmission) datagram (content of EOT datagram to be defined by you). Receiver will acknowledge every received datagram and save the data extracted from datagrams in file, using sequence numbers for proper reconstruction of the content of the file. File is delivered with no losses if it is an exact copy of the original file. At the end of transmission (when EOT is acknowledged) the sender will report Total-Transmission-Time.

The <reliability number> RN is used by the sender to emulate lost packets. Essentially in every group of RN packets in a sequence of sent packets one packet is not sent at random (thus emulating packet loss). If RN = 1 all packets are lost. If RN = 2 every second packet is "lost". If RN = 0 no packet loss is emulated.

ACKs sent by receiver do not require the simulation of losses.

For data packets in case of Go-Back-N, SeqNum is the modulo 128 sequence number of the packet. The sequence number of the first packet should be zero. For ACK packets in case of Go-Back-N, SeqNum is the sequence number of the packet being acknowledged.

Submission:

Submit two applications (with sources) – sender and receiver in separate zip files containing Java files only. DO NOT SUBMIT ECLIPSE PROJECTS! All Java files submitted have to use default package (no package keyword). It should be possible to compile your project in any command line environment with "javac *".

Submit 1 page document that describes the format of your datagrams, and also report on timing for transfer of small (under 4 Kb) and large (between 0.2 and 2 Mb) files using Stop-and-wait and Go-Back-N protocol for values of window size 10, 40 and 80, RN=0,5,100; and different values of timeout.