

Instructions for JNW2 Project TRN1 – Reliable Transport Layer

See chapter 10 of *Understanding Internet Protocols*. Project TRN1 is basically the same as given at the end of Chapter 10. Use the `JNW2.gui.GUIController` main class and load configuration file “ReliableTransport.txt”. Then run the simulation.

Your assignment is to complete `SendSegments.java` based on the algorithms in the comments it contains, which are essentially the same as those in *Understanding Internet Protocols*. Your assignment is to code Steps 1 and 3 of `sendRtlSegments()`. Remember to comment your code completely.

To understand what is needed, first study carefully `Segments.java` to learn how the JNW2 reliable transport works. The functions provided there are important for solving the assignment. The maximum segment is quite small: 20 bytes, in order that you can see how a TCP-like protocol works without having to grapple with large chunks of data.

Note that in TRN1 the NW protocol does not perform congestion control by varying its send window as TCP does, nor does it support multiple data flows simultaneously. Also it works with a fixed “email” message; the application can’t add data after a connection is made the way it is possible with TCP.

You should ignore this in part 3 of the algorithm: “also use non-negative `segmentSendTime` for sequenceNumber matching reply `ackNumber` to update RTT (use simple difference, not algorithm with 4D)” because `Segments()` does not propagate the information needed to do that.

When TRN1 is working correctly, two “email” messages will be sent: one from node 1.1 (router1) to 7.1 (router7), the other from node 1.1 (router1) to 2.1 (router2).

Submit your `SendSegments.java` and a copy of the output produced when you run the simulation with the `ReliableTransport.txt` configuration. Be sure to include your name in the code comment provided.

QUESTIONS TO BE ANSWERED AS PART OF JTRN1 SUBMISSION

Include your answers at the end of `output.txt`.

- What would you expect to happen if the JNWS TCP receiver were to reply with an ACK where the value of `acknowledgementNumber` is always zero?
- Suppose we were going to add a congestion window to JNW2 Reliable Transport. What variable in the current code could be used to impose a congestion window in `Segments.java`?
- If we had slow-start congestion avoidance in JNW2 Reliable Transport, what should be the window size at startup?