Project 2: The transport layer and ARQ implementation

*Due: Until finishing network layer lectures*

In this project, you will implement two well-known ARQ protocols, stop and wait and selective repeat, which the latter is ***optional*** with a big bonus point!. As these protocols have been explained in lectures, there is no need for further discussion. Instead, here we will denote some implementation facets.

# Segments and Acks Structure

1. Sequence number:
   1. For Stop and Wait 1-bit.
   2. For Selective Repeat 32-bit
2. 16-bit as segment indicator, which is 0101010101010101
3. 16-bit as ack indicator, which is 1010101010101010
4. 16-bit checksum in both segments and acks
5. Payload with a variable size, which we will denote as MSS

# Stop and Wait Specification

Stop and wait ARQ has multiple variations, but you will implement the most advanced version in this project. The sender FSM of this stop and wait ARQ is introduced in figure 3.15 as rdt3.0 (page 245 of the reference book. Edition 6). By looking at this FSM, we can see that this ARQ encounters noisy channels that packets can be lost or corrupted while passing through it. Do yourself a favor and draw the receiver side of this FSM before implementing it.

Summary of specifications:

1. Use timers for handling packet loss
2. Use checksum to check whether receiving packet/ack is corrupted or not.
3. Use sequence number as its obvious why.
4. Add segment indicator to your segments.
5. Add ack indicator to your acks.

# ***Selective Repeat Specification***

As you must know by now, this ARQ is far more effective than stop and wait ARQ. Everything seems good in theory about this ARQ, but it can be challenging to implement it in practice, and this is the only reason why this ARQ is not required in this project. To implement selective repeat, you have to be familiar with threads and deal with condition races. We will not explain it because you have to implement it word by word according to the book.

# Simulation and Results

1. Simulate packet loss with a probability of P. You will get a P where from the command line (as one of your arguments). Then, upon receiving packets in the receiver, generate a random number . Whenever discard the received packet.
2. Alter one bit (choose this randomly) with the probability of 0.01 at sender using the exact mechanism as the above statement.
3. You have to prepare average results for submission. For each of the following parts, you have to repeat the simulation five times, then report the average of those iterations in your report.
   1. Simulate delay over MSS. MSS starts from 100-bytes and goes up to 1000-bytes in 100-bytes steps.
   2. *Simulate delay over window size. Window size starts from 1 and goes up to 64 doubling at each step.*
   3. Simulate delay over P. P starts from 0 and goes up to 0.2, and increases by 0.01 at each step.
4. Plot the result of your simulations and include them in your report.

# Implementations Note

1. Use UDP for both cases.
2. You will be sending a file over these ARQs. (generate a simple text file).
3. Get Window size, MSS, P, and filename from arguments (Don’t hard code them).

# Submission

1. ***An essential factor in this project*** (as your second project) ***is to write readable and clean codes. We wanted you to be aware of this in advance.*** (there is a lot of material about this in the literature of software engineering)
2. You are **not allowed** to use any non-standard python library in the project. (non-standard libraries are the ones you have to download them to use)
3. Include a **self-explanatory** report in submission.
4. In your report, explicitly state which parts of the project have been implemented and which have not.
5. Include any kind of information that you think will help us better understand what you have done. Like figures, diagrams, etc.
6. Start early.
7. Groups are **not allowed**.
8. We will consider bonus points for projects using Object Oriented Programming (OOP) in their implementations.
9. Include a **complete walk-through** for running the project in your report. (You’ll not get any point if you don’t include this walk-through)

# Update 1

1. You can use the Matplotlib library, which is not a standard library in python, for plotting your results.
2. In the Simulation section and subclause 3. b, the delay is defined as the time it takes ARQ to transmit the sample file completely.
3. In the Simulation section and subclause 2, you have to alter a bit after calculating the checksum of the segment.