RDT 2.2

Implementation

The first packet to be sent will have sequence number of 0 and to handle the flipping of sequence number in the presence of corruptions we saved an attribute to sender and receiver sides called self.sequence which starts with 0 and only gets flipped when the data is not corrupted which means it is always the correct reference to compare with the data.

SENDER SIDE

Explanation of functions

get_checksum(data):

Get ASCII code of given data which is equivalent for us to the checksum of data.

is_corrupted(reply):

Checks if reply is corrupted. Returns true (when corrupted) if acknowledgment is any other number than 1 and 0 OR the checksum of the reply is not equivalent to the checksum of the acknowledgment by using get_checksum.

Is_expected_seq(reply,exp_seq):

Checks if the acknowledgment of the reply is as expected.

We are attempting to send packets saved in the buffer of the sender side to the receiver side for each packet in the sender buffer we clone the packet so that if it got corrupted we still have the original packet.

1-We send the clone packet to receiver and receive reply message from receiver

2-We check the reply message received if corrupted using function is_corrupted(reply) and also check if the acknowledgment received is the one expected or not using function self.is_expected_seq(reply,self.sequence)

if reply corrupted or the acknowledgment received is NOT as expected print message indicating corruption and resend the clone packet again

If reply is not corrupted and acknowledgment received is as expected flip the sequence number and break the loop for this given packet so that the next packet start being sended.

RECEIVER SIDE

Explanation of functions

is_corrupted (packet):

Checks if packet is corrupted. Returns true (when corrupted) if checksum is any other number than 1 and 0 OR the checksum of the packet is not equivalent to the checksum of the data by using ord () —Returns ASCII code

Is_expected_seq (rcv_pkt, exp_seq):

Checks if the sequence number of the received packet is as expected.

We are attempting to receive packets from sender to be saved in a buffer if the packet is not corrupted.

1-Receiver receives packet checks if packet is corrupted using is_corrupted(rcv_pkt) and also check if sequence number is the one expected or not using function self.is_expected_seq(rcv_pkt,self.sequence)

If received packet is corrupted or sequence number is NOT as expected

Print message indicating corruption and send reply to the sender with wrong acknowledgment so that sender side knows that an error occurred in past packet (i.e. send negative acknowledgment)

If received packet is not corrupted and the sequence number is as expected

Send reply with correct acknowledgment

Flip the sequence number for the next packet

Save the packet inside the receiver buffer