Name: Kshitij D
Roll no: CS17B104
Assgn no:2
Title: Go_Back_N Protocol

We can see that if packet drop probability is low, then, the retransmission ratio is low, and consequently, our programs end with lesser number of iterations. UDP sockets have being applied here. Here, if N is window size, then we take log2(N+1) to be the sequence number length, and convert to binary the remainders got by dividing successive incremented counts, by N+1. Ex:
If N+1=4. The, 000, 001, 010, 011 are the 4 values.
Packet length is a uniform distribution from 256 bits to the specified max_packet_length.

Observe that if timer of sender runs out before sender has sent packets, then, in receiver side, we have to change the expected sequence number.

Avg RTT calculation is: [Previous RTT*Total_no_of_packets_sent_previously+Sum across all packets sent now(Time for each to return)]/(Total_no_of_packets_sent_previously+Packets_sent_just_now)

RTT value increases in general after every iteration.

We keep window at receiver side as 1. Note that by GBN, our bandwidth usage is much better than Stop_and_wait protocol. Retransmission ratio always>1.
Cumulative ack is being used, in that, if an unexpected packet comes, then receiver knows that the expected packet has defaulted and sends an ack with a value corresponding to the expected packet's, so that the sender knows that this packet has being lost, and it can begin retransmitting every packet from his window since this point.