Assignment 6.3 RDT Implementation

In this assignment, you need our given UDP socket class to implement RDT protocol. Then implement an Echo Server and Client.

I. Requirement

i. Your protocol needs to ensure the reliability of data transfer. Packet loss and payload corruption might happen.

i. To deal with packet loss, using ack and retransmission according to GBN the textbook.

ii. To deal with payload corruption, you need to design a checksum of your payload.

ii. Payload Example

Your payload might be like this:

| SYN | FIN | ACK | SEQ | SEQ ACK | LEN | CHEKCSUM | Payload|

| 1 bit|1 bit| 1 bit| 4 byte| 4 byte | 4 byte | 2 byte | LEN |

i. Checksum Calculation Example

```
def calc_checksum(payload):
    sum = 0
    for byte in payload:
        sum += byte
    sum = -(sum % 256)
    return (sum & 0xFF)
```

II. API reference:

rdt code example:

```
from udp import UDPsocket # import provided class
class socket(UDPsocket):
    def __init__():
        super(socket, self).__init__()

    def recvfrom():
        # your code here
        pass

    def sendto():
        # your code here
        pass
```

server code example:

```
from rdt import socket

server = socket()
server.bind((SERVER_ADDR, SERVER_PORT))
while True:
    dara, addr = server.recvfrom()
    server.sendto(data, addr)
```

client code example:

```
from rdt import socket

client = socket()
client.send(DATA, (SERVER_ADDR, SERVER_PORT))
data = client.recvfrom()
assert data = DATA
client.close()
```