Network Project CSEN503

**RDT 2.2 Report**

**1-Team members Info:**

Mohamed Elsaeed Mohamed Elmenshawy 55-12559 T-12

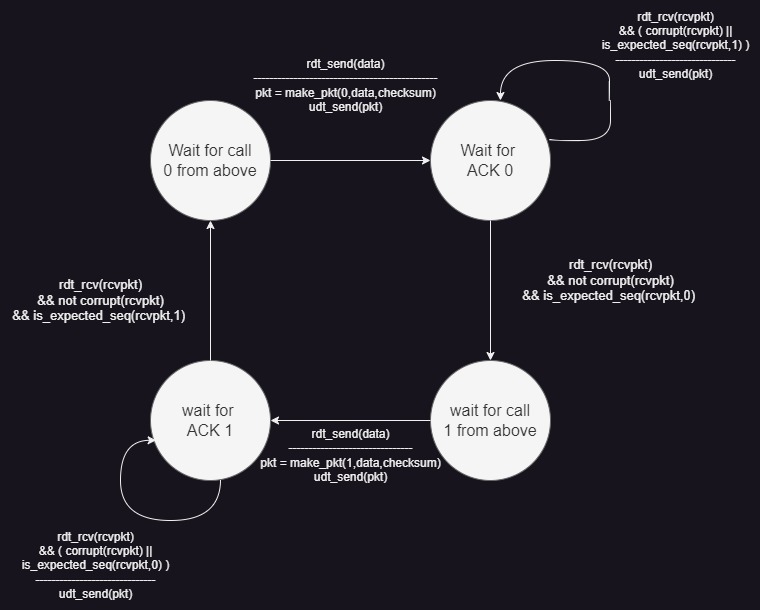
Mahmoud Hany Hebishy 55-18387 T-11

**2-Roles of each member :**

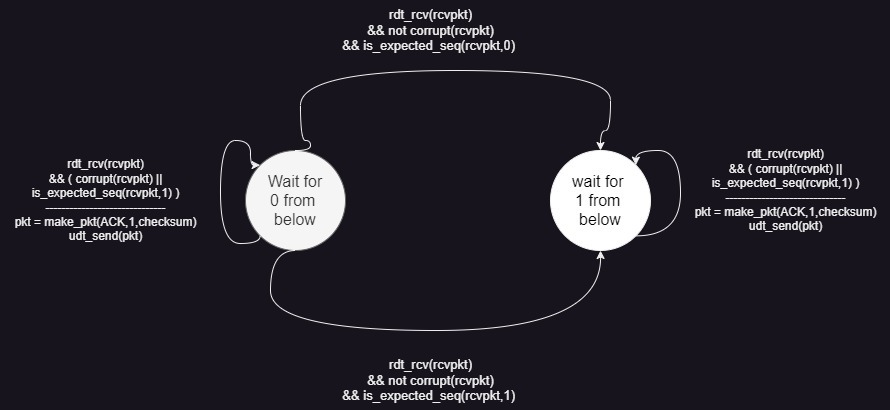
1. Mohamed : was responsible for the receiver side and handling the corruption of packet received also was responsible for handling some logic in the sender side
2. Mahmoud : was responsible for the sender side and handling the corruption of reply received also was responsible for handling some logic in the receiver side

**3-FSM diagrams:**

1-Sender side:

****

2-Receiver side:

****

**4-pseudo-code for RDT sender and receiver :**

1-Receiver Side :

1-First we need to check if the received packet from sender Corrupted or if its sequence number not equal to the expected one

2-IF TRUE

a)Show the corrupted Packet

b)Change the sequence Number

c)Make a corrupted Packet with the changed Sequence number

d)Print the expected sequence number from receiver

e)Print the corrupted Packet

d)Return the corrupted Packet

3-ELSE

a)add the correct data delivered to the puffer

b)Make a new Packet with the same sequence number received from sender

c)Print the expected sequence number from receiver

d)Print the new Packet

e)Return the new Packet

2-Sender Side :

1- We Loop till the last character in the buffer

a) We get the Unicode of the current character

b) Create new Packet with this value

c) we Clone this Packet and save it

d) Print The sequence number of the current packet

e) Print the Packet

f) Send The Packet to the receiver

g) check for the corruption for reply from the receiver or the sequence number of the reply not the expected sequence

i)IF TRUE

1)Loop till the packet sent successfully

a)Send the cloned packet

h)change the sequence number to detect next character

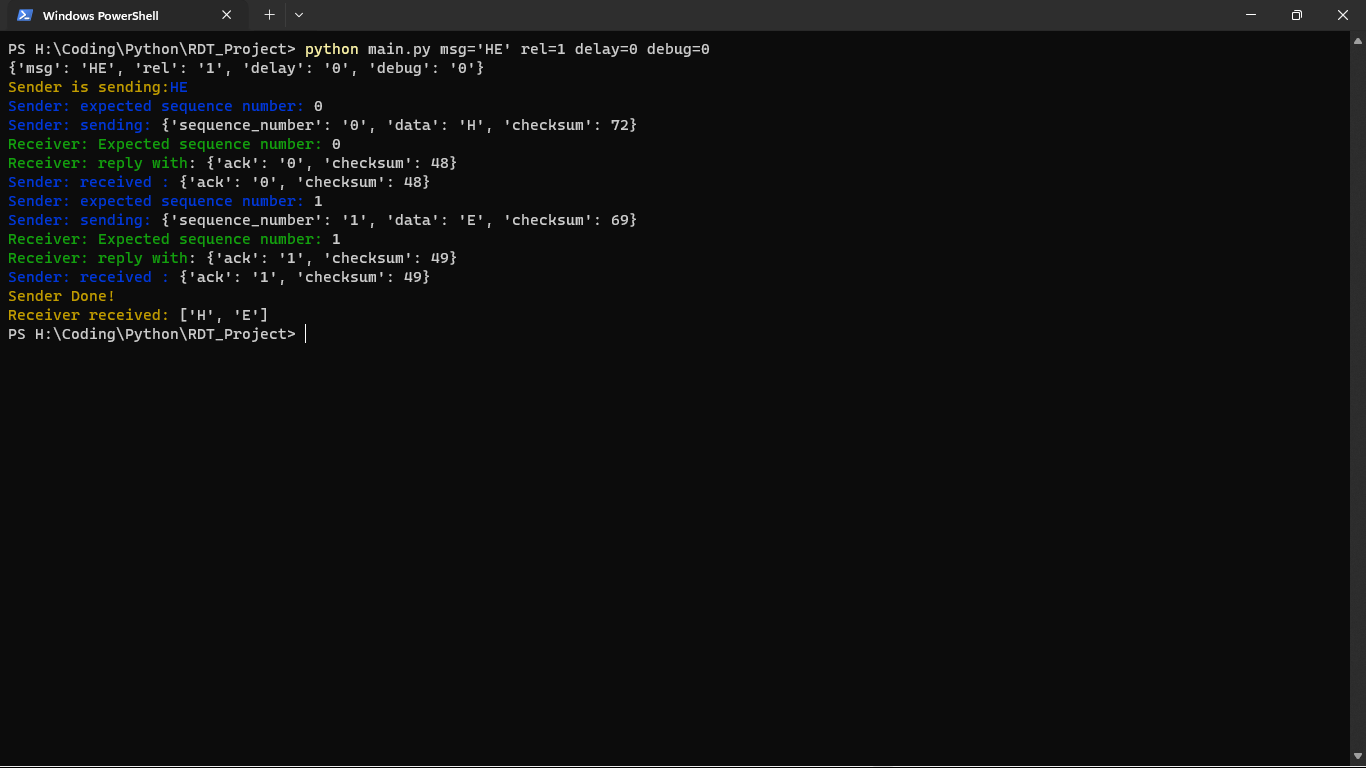
2- Print Sender Done

**5-Addition to network.py file**

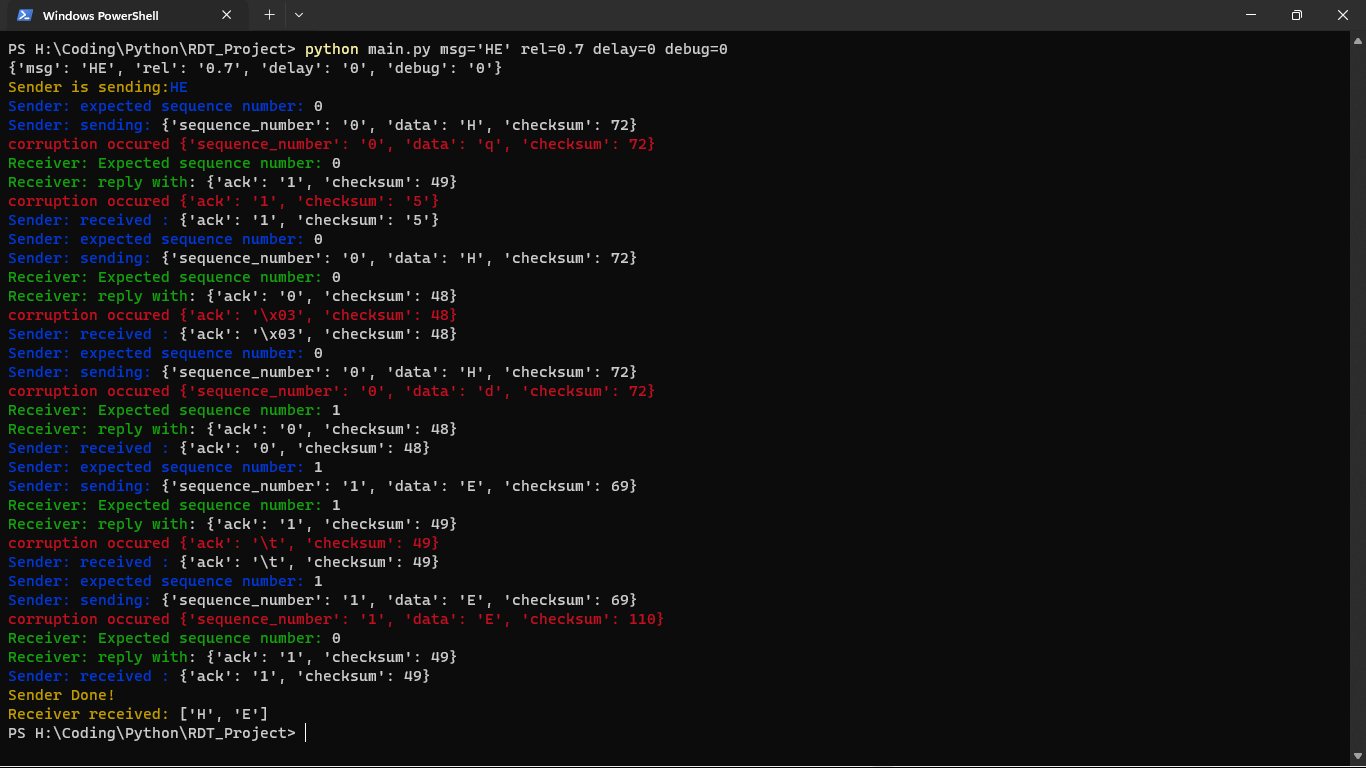
We printed the corruption packets or reply only if happened in “udt\_send” method

**6-Test cases**

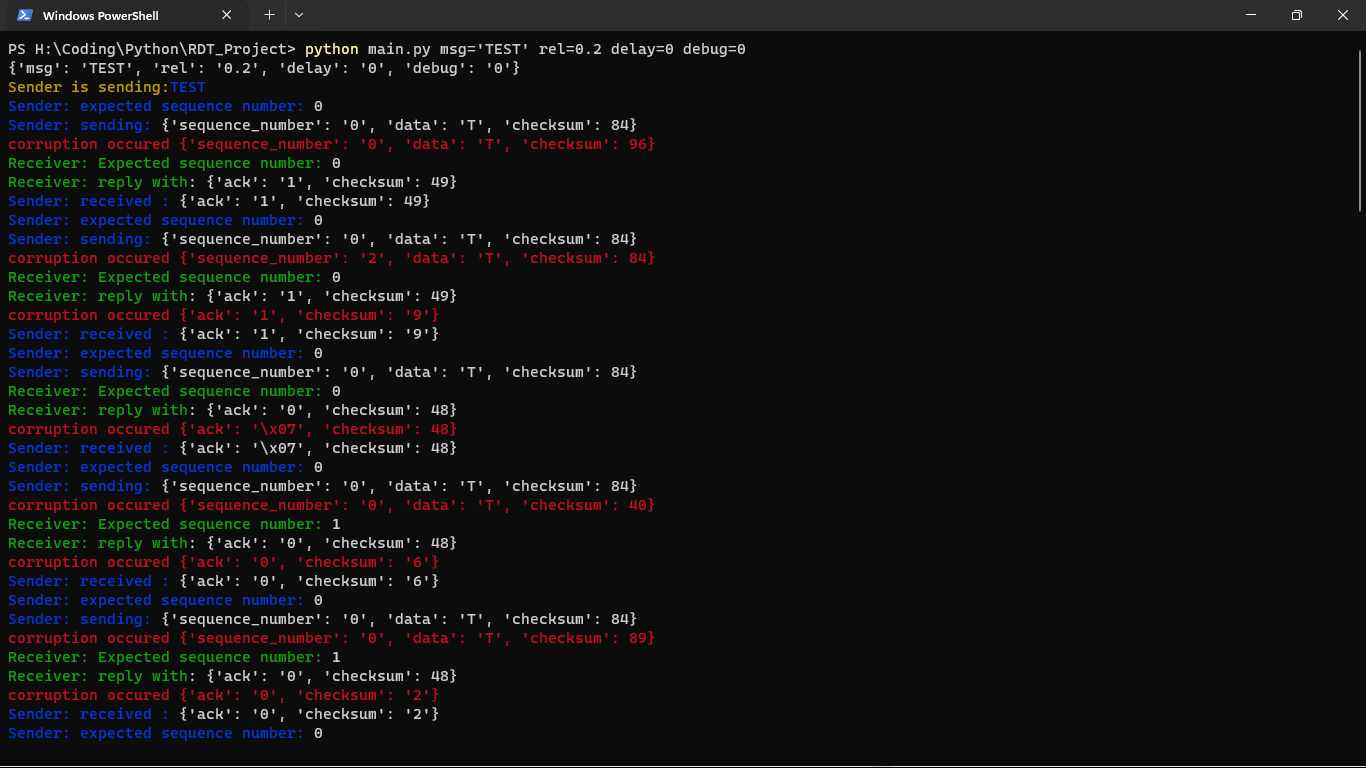
1. Sending “HE” With reliability =1

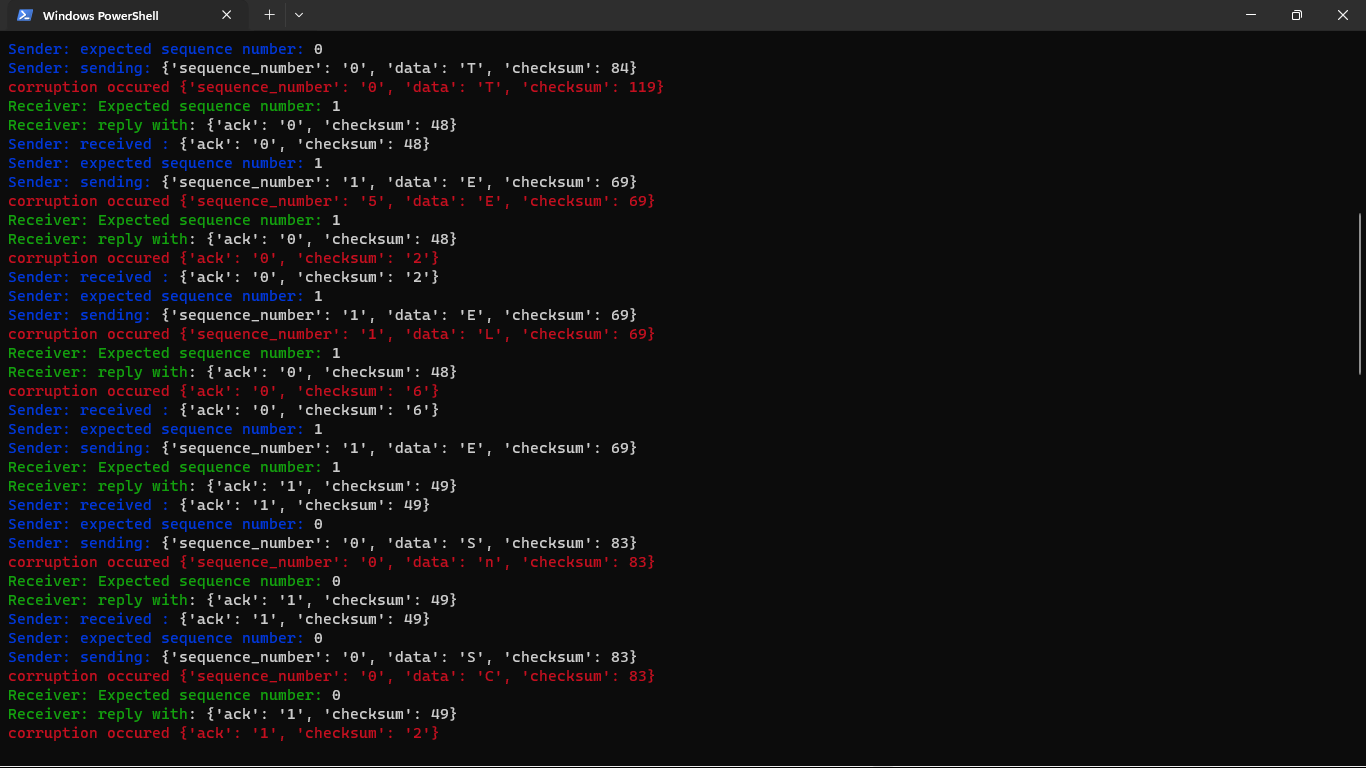


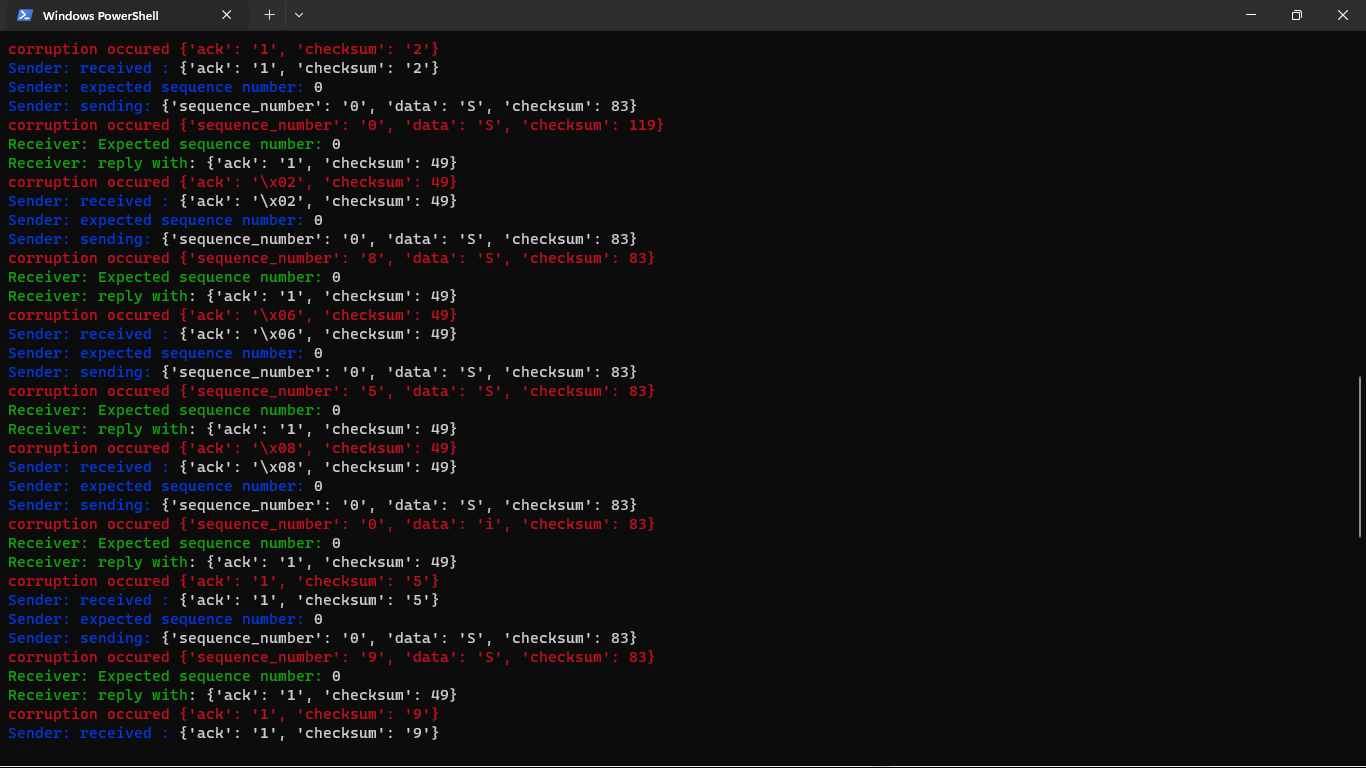
1. Sending “HE” With reliability =0.7

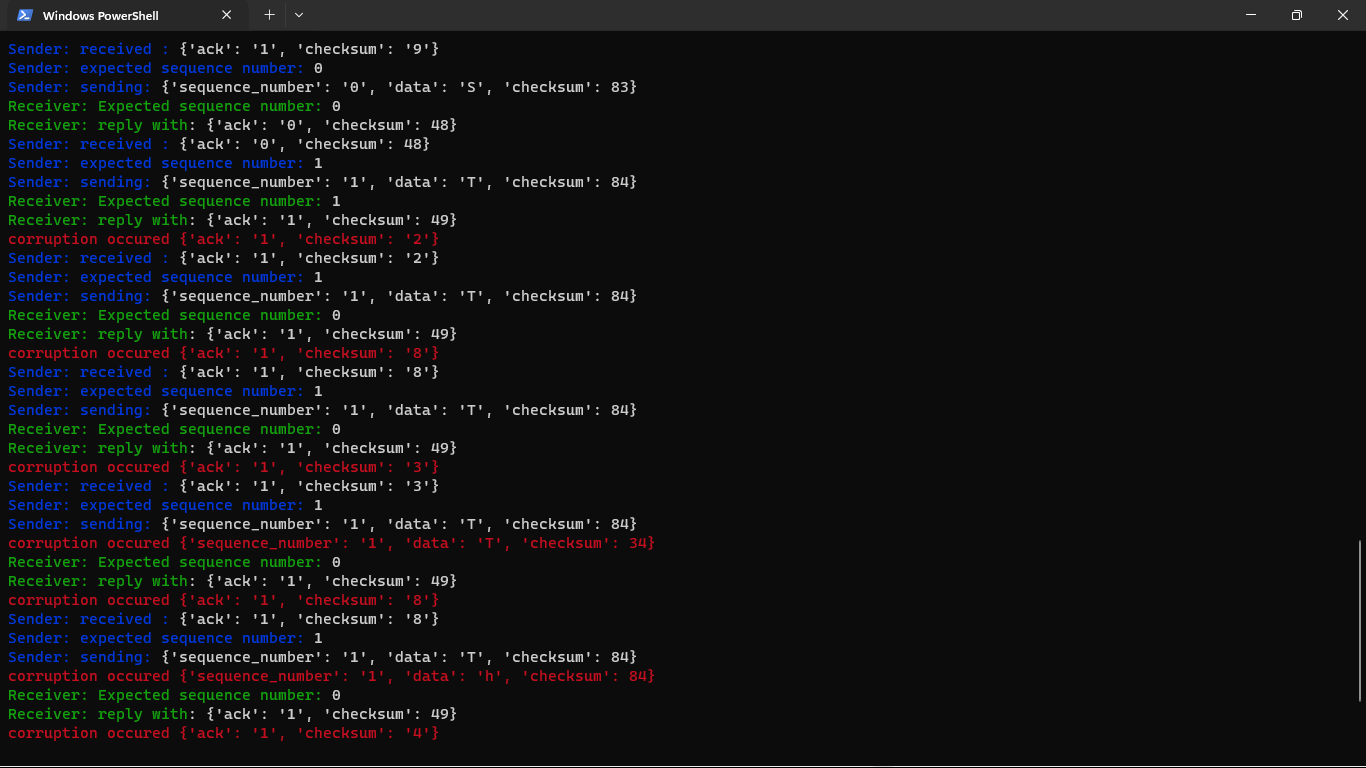


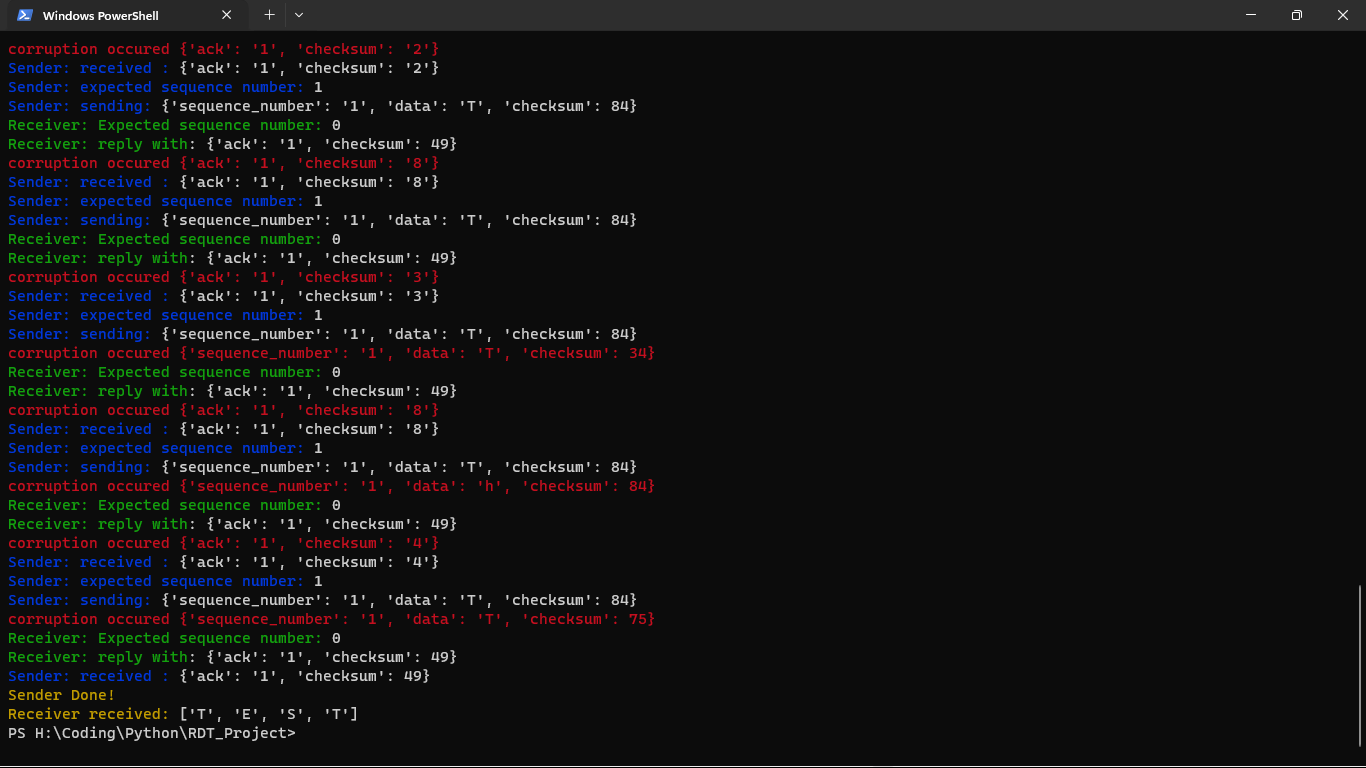
1. Sending “TEST” With reliability =0.2











**Conclusion:**

In conclusion, it can be observed that the transmitted message successfully reaches the intended receiver in all cases, fulfilling the primary objective of RDT v\_2.0.