

Giselle Northy

Project 2 - Reliable Data Transmission (RDT)

Nov 2, 2020

Introduction

- In this coding project we started with skeleton code for an RDT layer that allows the transfer of string data through an unreliable channel in a simulated environment. Code for the unreliable channel, the Segment class, and also the calling main function, is provided. Essentially we had to build out and fill out missing functionality in the code.

How it works

- Used skeleton code template version 1 on Windows machine
- The skeleton code provided creates a client to send data to a server
- Both are running class **RDTLayer**.
- The client first manages sending and manages receiving. The server does the same.
- In the **manageSend** function, the client checks to see if any segments have timed out and will resend if needed.
- Then it will send segments broken up into **DATA LENGTH** which is 4 characters out that fit in the **FLOW CONTROL WINDOW** which is a constant of 15 characters.
- On **manageReceive**, it accepts all incoming messages from the channel
- If the sequence number is greater than the previous ACK number it will save the packet for processing later as long as the packet's checksum passes.
- If the sequence number matches the previous ACK number, it knows this is the next segment we are looking for.
 - It checks the packets Checksum to verify the data is not corrupt.
 - Saved packets are checked and added to the data received, if they are next in order.
- One ACK message is sent for an incoming set of messages as a cumulative ACK.
- If the incoming packet is an ACK message, all packets with sequence numbers less than the ACK are cleared for timeout.

How to run

- Run the program from terminal Python from **rdt_main.py** for skeleton code version 1
- Example of the run command in a Windows terminal:
 - **py -3.7 rdt_main.py**

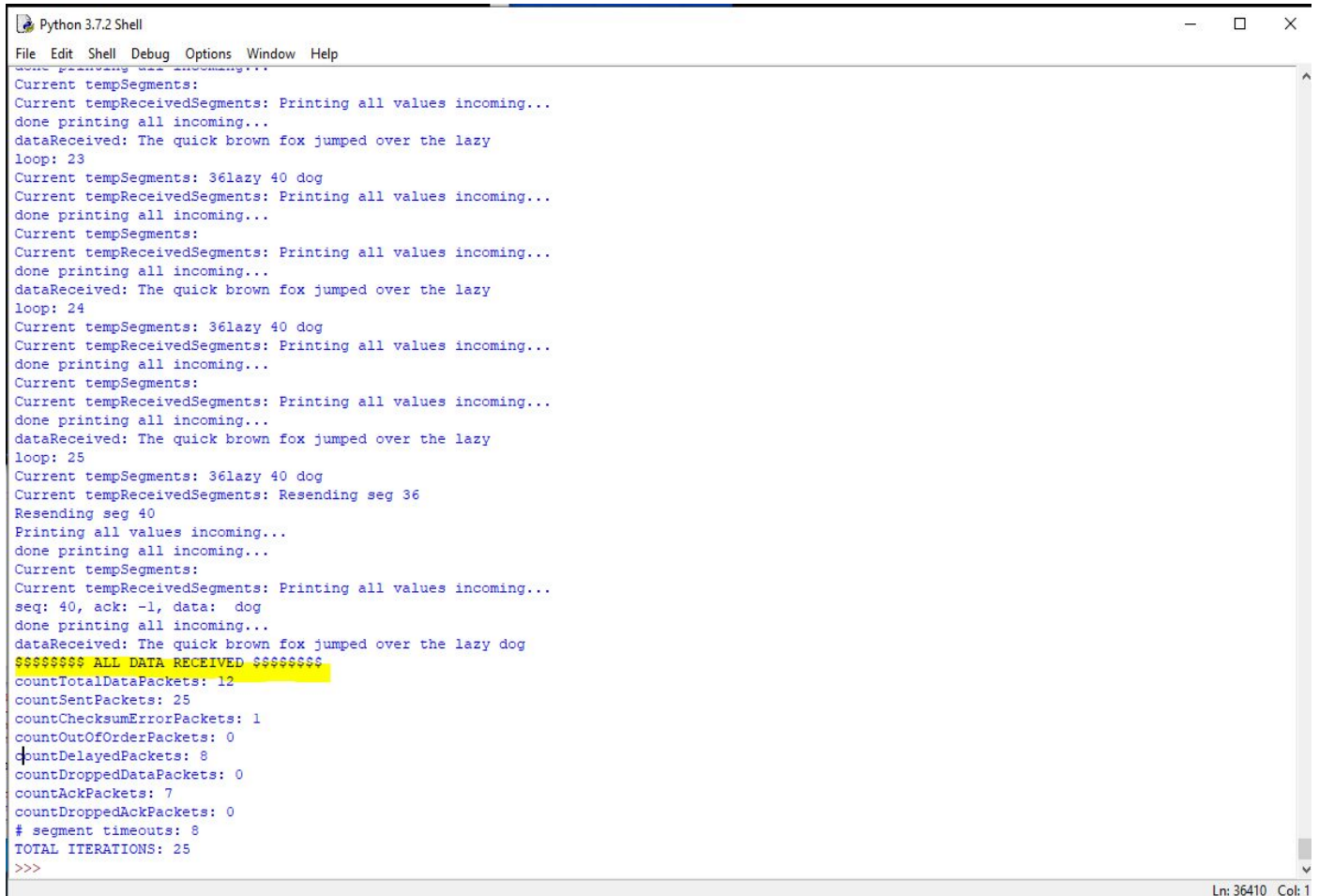
Comments

- I found this assignment to be very challenging. The concepts from the lectures related to RDT made sense, but finishing the Python skeleton code was hard. Partly I think this is because I have not programmed in Python much. The Piazza discussions were helpful and gave me some good ideas on how to get started.

Screenshots of running code

Example A) Running the shorter string “The quick brown fox jumped over the lazy dog”

Note: Included extra print statements to keep track of current segments



```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
done printing all incoming...
Current tempSegments:
Current tempReceivedSegments: Printing all values incoming...
done printing all incoming...
dataReceived: The quick brown fox jumped over the lazy
loop: 23
Current tempSegments: 36lazy 40 dog
Current tempReceivedSegments: Printing all values incoming...
done printing all incoming...
Current tempSegments:
Current tempReceivedSegments: Printing all values incoming...
done printing all incoming...
dataReceived: The quick brown fox jumped over the lazy
loop: 24
Current tempSegments: 36lazy 40 dog
Current tempReceivedSegments: Printing all values incoming...
done printing all incoming...
Current tempSegments:
Current tempReceivedSegments: Printing all values incoming...
done printing all incoming...
dataReceived: The quick brown fox jumped over the lazy
loop: 25
Current tempSegments: 36lazy 40 dog
Current tempReceivedSegments: Resending seg 36
Resending seg 40
Printing all values incoming...
done printing all incoming...
Current tempSegments:
Current tempReceivedSegments: Printing all values incoming...
seg: 40, ack: -1, data: dog
done printing all incoming...
dataReceived: The quick brown fox jumped over the lazy dog
$$$$$$$ ALL DATA RECEIVED $$$$$$$
countTotalDataPackets: 12
countSentPackets: 25
countChecksumErrorPackets: 1
countOutOfOrderPackets: 0
countDelayedPackets: 8
countDroppedDataPackets: 0
countAckPackets: 7
countDroppedAckPackets: 0
# segment timeouts: 8
TOTAL ITERATIONS: 25
>>>
```

Ln: 36410 Col: 1

Example B) Running the “long” string “...We choose to go to the moon...”

```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help

Current tempReceivedSegments: Resending seg 1236
Resending seg 1240
Client RDTlayer
Printing all values incoming...
done printing all incoming...
Current tempSegments:
Current tempReceivedSegments: 12402
Server RDTlayer
Printing all values incoming...
seq: 1236, ack: -1, data: 196
seq: 1236, ack: -1, data: 196
seq: 1240, ack: -1, data: 2X
done printing all incoming...
dataReceived:


...We choose to go to the moon. We choose to go to the moon in this decade and do the other things, not because they are
easy, but because they are hard, because that goal will serve to organize and measure the best of our energies and skills
, because that challenge is one that we are willing to accept, one we are unwilling to postpone, and one which we intend
to win, and the others, too.

...we shall send to the moon, 240,000 miles away from the control station in Houston, a giant rocket more than 300 feet t
all, the length of this football field, made of new metal alloys, some of which have not yet been invented, capable of st
anding heat and stresses several times more than have ever been experienced, fitted together with a precision better than
the finest watch, carrying all the equipment needed for propulsion, guidance, control, communications, food and survival,
on an untried mission, to an unknown celestial body, and then return it safely to earth, re-entering the atmosphere at sp
eeds of over 25,000 miles per hour, causing heat about half that of the temperature of the sun--almost as hot as it is he
re today--and do all this, and do it right, and do it first before this decade is out.

JFK - September 12, 1962

$$$$$$$$ ALL DATA RECEIVED $$$$$$$$
countTotalDataPackets: 464
countSentPackets: 646
countChecksumErrorPackets: 37
countOutOfOrderPackets: 21
countDelayedPackets: 71
countDroppedDataPackets: 51
countAckPackets: 176
countDroppedAckPackets: 14
# segment timeouts: 207
TOTAL ITERATIONS: 657
>>>
```

Example C) Running in Command Line

 Select Command Prompt

```
Current tempSegments: 28ver 32the 36lazy
Current tempReceivedSegments: Printing all values incoming...
seq: -1, ack: 40, data:
done printing all incoming...
Current tempSegments:
Current tempReceivedSegments: Printing all values incoming...
done printing all incoming...
dataReceived: The quick brown fox jumped over the lazy
loop: 31
Current tempSegments:
Current tempReceivedSegments: Printing all values incoming...
done printing all incoming...
Current tempSegments:
Current tempReceivedSegments: Printing all values incoming...
seq: 40, ack: -1, data: dog
done printing all incoming...
dataReceived: The quick brown fox jumped over the lazy dog
$$$$$$$$ ALL DATA RECEIVED $$$$$$$$
countTotalDataPackets: 21
countSentPackets: 28
countChecksumErrorPackets: 3
countOutOfOrderPackets: 1
countDelayedPackets: 1
countDroppedDataPackets: 3
countAckPackets: 10
countDroppedAckPackets: 1
# segment timeouts: 11
TOTAL ITERATIONS: 31

E:\SCHOOL\CS\CS372\project_2\skeleton_code.v01>py -3.7 rdt_main.py
```