STEPS:

- **Analyse a trace of the TCP segments sent/received when transferring a 150KB file from computer to remote server**
    - TCP's use of sequence and acknowledgement numbers for providing a reliable data transfer (RDT)
    - TPC's congestion control algorithm (slow start and congestion avoidance)
    - TCP's receiver-advertised flow control mechanism
    - TCP's connection setup and performance (throughput + round-trip time) between computer and server

CAPTURING A BULK TCP TRANSFER FROM COMPUTER TO SERVER

a. Use wireshark to obtain packet trace of TCP transfer of a file from computer to server
b. Access a web page to enter the name of a file stored on computer
c. Transfer file to a webserver using the HTTP POST method (requests for web server to accept data) from computer to another computer

http://gaia.cs.umass.edu/wireshark-labs/alice.txt

http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html.

a. Use browse button- enter name of file
b. Don't upload yet
c. Start up wireshark and begin packet capture
d. Go back to browser and upload file to server
e. Stop wireshark packet capture
f. Filter to TCP

CAPTURED TRACE

a. See initial three-way handshake containing a SYN message
    - Multiple HTTP continuation messages- indicates there are multiple TCP segments used to carry a single HTTP message
    - TCP ACK segments returned from gaia.cs.umass.edu to computer

b. Open file tcp-ethereal-trace-1 in wireshark
c. Print packet: file>print selected packet only, choose packet summary line, select minimum amount of packet detail

227…	32.073583	172.19.126.23	52.168.117.175	TCP	1440	50887 → 443 [ACK] Seq=217466 Ack=713 Win=2048 Len=1374 TSval=1433264269 TSecr=1152034116 [TCP segme
227…	32.073585	172.19.126.23	52.168.117.175	TCP	1440	50887 → 443 [ACK] Seq=218840 Ack=713 Win=2048 Len=1374 TSval=1433264269 TSecr=1152034116 [TCP segme
227…	32.073586	172.19.126.23	52.168.117.175	TCP	1440	50887 → 443 [ACK] Seq=220214 Ack=713 Win=2048 Len=1374 TSval=1433264269 TSecr=1152034116 [TCP segme

1.

ip.src == 172.19.126.23 (my computer)
TCP Src Port = 50887

2.

ip.dst == 52.168.117.175 (gaia.cs.umass.edu)
TCP sending/receiving port: 80

3.

ip.src == 172.19.126.23 (my computer)
TCP Src Port = 50887

4.

Connection initiation is TCP SYN segment, sequence number = 0 (handshake- relative sequence number: initiates TCP connection between client and computer)

5.

First packet sent will be TCP ACK segment, sequence number = 1

6.

HTTP POST- requesting for server to accept data
Sequence number: 151705
Sequence number of segment is 1

7.

Segment 1 sequence number= 151705, [iRTT: 0.286618000 seconds]

Segment 2 sequence number= Sequence Number: 631, [iRTT: 0.286618000 seconds]
Segment 3 sequence number= Sequence Number: 1387, [iRTT: 0.286618000 seconds]
Segment 4 sequence number= Sequence Number: 2017, [iRTT: 0.286618000 seconds]
Segment 5 sequence number= Sequence Number: 2773, [iRTT: 0.286618000 seconds]
Segment 6 sequence number= Sequence Number: 3403, [iRTT: 0.286618000 seconds]

Time sent
0.271257 0.271425 0.271797 0.271798 0.367081 0.368711

ACK received
0.366931 0.367289 0.368617 0.369952 0.479965 0.482492

8.

Length of first TCP segment = 631 bytes
length of each of the following five TCP segments = 756 bytes.

9. What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?

Minimum amount is 631- throttle doesn't occur
Throttle: enables transmission rate to slow down to emulate the experience of users

10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) to answer this question?
No retransmission- time rates are different for each sequential packets
Retransmission: sending data again

11. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 on page 250 in the text)
Receiver ACKs =1386 bytes

- No. 80 and 87 is when the process of cumulative ACKS occur.

12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.

Th = data size/ time

13. Use the Time-Sequence-Graph (Stevens) plotting tool to view the sequence number versus time plot of segments being sent from the client to the gaia.cs.umass.edu server. Can you identify where TCP's slow start phase begins and ends, and where congestion avoidance takes over? Comment on ways in which the measured data differs from the idealized behaviour of TCP that we've studied in the text.

No congestion control: low traffic + bandwidth is too large

14. Answer each of two questions above for the trace that you have gathered when you transferred a file from your computer to gaia.cs.umass.edu