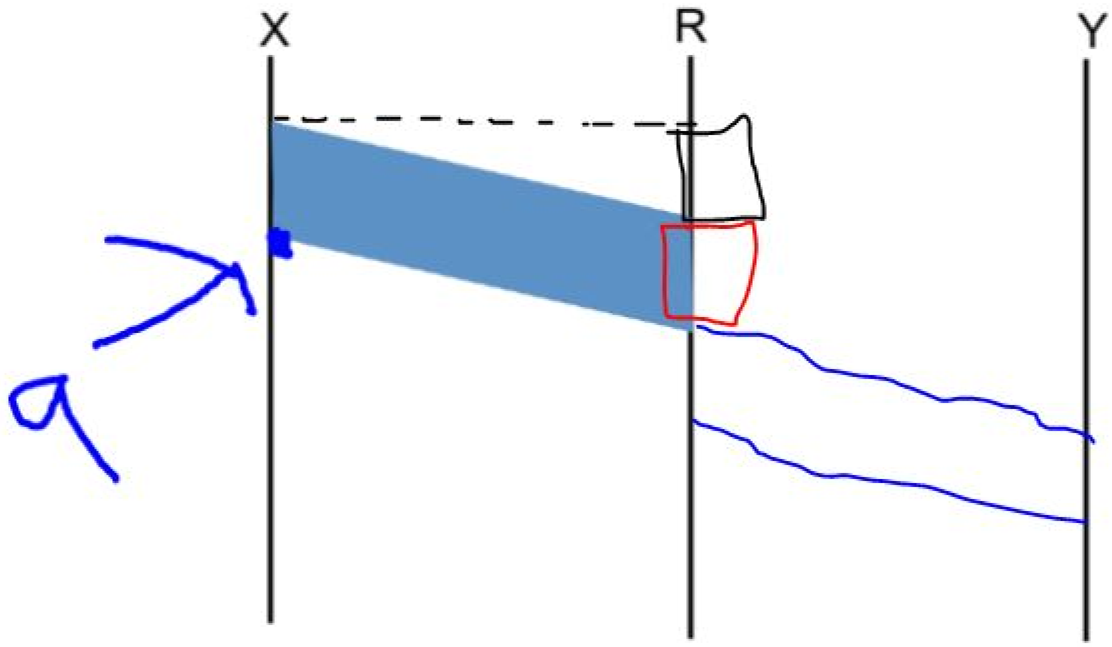
1



B. Transmission delay is the black square, while propogation delay is the red square.
c.The poorly drawn blue lines are a rough estimate of the completion of the time diagram.

2.

a. (5 + 6 + 7) / 3 = 6ms standard dev = 1

23 + 26 +23 / 3 = 24 ms standard dev = 2
14 + 14 + 14) / 3 = 14 ms standard dev = 0

b. First was 8 routers, second was 9, and the third was 11 different routers.
c. I see about 4ish difference groups of similar IP addresses. It seems that yes the longest delays appeared between certain networks, but other network changes seemed to have no real difference in delay. The jump to googles actual server (who I tested) seemed to be the biggest delay each time.
d. 71 + 74 + 72 / 3 = 72.333 ms standard dev = 1.5 So the standard deviation is still in line with the ones within the United States, but as expected the delay is much much longer than even my second call which took place at 8pm, when traffic was high.

3.
a. Dprop = m/s seconds
B. Dtrans = l/r seconds
C. De2e = (m/s + l/r) seconds
D. The last bit of the packet would be leaving Host A at that moment.
E. The first bit has left Host A, and is in the link somewhere, it hasn't yet reached Host B.
F. The first bit of the packet would have just arrived at Host B

g. Set the equation in A equal to the one in B. Solving for m yields ls/r. Plugging in the given numbers gives (500 * 2.5 * 10^8)/128000. The final answer is 976.6 kilometers.

4.

a. The message is 8000000 bits long, and the connection is 2000000mb/s so the time will be 4 seconds to each stop. 2 routers means 3 trips so 4 * 3 = 12 seconds total
b. First packet is 10000 bits, divide that by 2Mb/s again and that's .005 seconds at the first switch. For the second switch just double the time to .010 seconds.
c. Well continuing off of part B, to get to the destination will be .005 * 3 so .015 seconds. However following packets will already have been sent once it arrived at the previous router, so the next packet is already only .005 seconds away. So .015 + (.005 + 799) = 4.01 seconds. This is nearly three times faster than not using message segmentation
d. Errors can't be handled effectively without message segmentation. Resending a bit is easy, but resending the entire thing would take much longer every time there is a problem. Small packets also reduce overall delays as huge packets would delay things for everyone else, instead of more even delays when everyone uses a small packet size.
e. There will be a lot more overhead with header bytes, as the size of that is independent of the amount of packets, so each still needs one. The packets also need to be reassembled (in the same specific order) when they reach their destination, so it could potentially cause a delay. Though I'm not sure in what cases that delay would outweigh the benefits, I'm sure some exist.

5.

App Layer							M
Transport Layer					P1	P2	M
Network Layer			A1	A2	P1	P2	M
Link Layer	M1	MR1	A1	A2	P1	P2	M
Physical Layer	M1	MR1	A1	A2	P1	P2	M
Gets router Link Layer	MR2	M2	A1	A2	P1	P2	M
Data arrives to	MR2	M2	A1	A2	P1	P2	M

Host							