COMP90007 Internet Technologies: Lab 1 Solution

1 Question 1

Several ways to do this, but you can use 'ipconfig' on the command line (on Windows) to determine your IP address; or from Wireshark itself, determine that it is your computer sending the HTTP GET request to request a web page from the remote server.
The source IP address will depend on your computer. The destination IP address for the HTTP based website can be confirmed by typing the IP address in a web browser and hitting enter.

2 Question 2

The amount of traffic captured will depend on the network conditions, whether some packets were lost and needed to be retransmitted, whether some background network traffic was also captured, etc.

3 Question 3

Again results may vary, but the overhead (for our purposes, which includes the headers from each of the packets and also the traffic involved with establishing and ending the connection) can be obtained by dividing the total traffic in bytes received from the previous question, by the file size of the retrieved HTML file (which is 44,220 bytes when I tried).

4 Question 4

The overheads/headers are used to allow each layer in the networking architecture perform their particular service. For example, the networking layer headers will have the source IP and destination IP addresses for that packet, which tells the intermediate hosts where the final destination of the packet is intended to be.
These headers are part of how data is transferred between the networking layers, what we call "encapsulation", and keeps networking functions separate and modular between the layers.
Arguably, the size of the information contained in the headers can be reduced by having a "monolithic" architecture to networking, where there is only one "layer", but then modularity and abstraction are sacrificed, which is bad from an engineering perspective.

5 Question 5

Apart from HTTP GET and HTTP 200 OK request and response packets, other packets of interest include the SYN and SYN ACK packets at the beginning which establishes a reliable

connection between two hosts (more on that when we get to TCP), and also the packets at the end when the connection is closed.
The packets in the middle are the data that make up the rest of the actual web page. Because the web page is quite large, the data will be split and transferred with multiple packets. It can also be observed that between every two or three packets that are received, our computer sends an ACK or acknowledgement packet to communicate to the remote host that the packets sent were successfully received. If the remote host fails to receive these packets, then it can be assumed that they were lost in transmission and need to be resent.
We will cover TCP handshaking and maintaining a reliable method of communication between two hosts in more detail as we progress in the semester.

6 Question 6

Similar to the last part in Question 4, a non-layered architecture may be more efficient, but at the cost of flexibility and modularity. Other disadvantages include a sizable overhead if the information to be transferred needs to be split across many packets.
However, the advantages of modularity and abstraction (also in engineering overall) outweigh the disadvantages. Advantages include, but aren't limited to, information hiding (i.e. if you were a network engineer, you would not need to know detailed information on the physical layer to work on improvements to another layer), and flexibility (the ability to change protocols in a certain layer, without affecting the operation of other layers).
There is a topic in research called "cross-layer optimisation" which breaks the barrier between the layers in an attempt to squeeze more performance, but this goes against the principle of abstraction.

7 Question 7 (Bonus Question)

For this question, you would have needed to plot a histogram, i.e. on the x-axis is a "bin" for a certain range of IATs, and on the y-axis the frequency or number of packets which fall into each particular bin. You should observe an exponential distribution. Feel free to use your favourite scientific computing/plotting package (Excel, MATLAB, Python, Mathematica, pen and paper, etc.).