Please answer each question with a written response (no code required). Feel free to use whatever software you want for your write-up. However, you should convert it to a `pdf` file for your submission upload.

Question 1: Fetching HTTP Resources

In section 2.2.1, the book talks about fetching resources from a HTTP server via HTTP requests. Let's say I wanted to fetch and load an `html` file called `index.html`. This file includes another `html` file called `header.html` that must also be fetched from the server. Additionally, there are two images files that must be fetched: `portrait.png` included in `index.html` and `logo.png` included in `header.html`. These resource dependencies can be views as a tree as follows:

```
                          index.html 5240B


        header.html 1200B          portrait.png 16000B


             logo.png 8400B
```
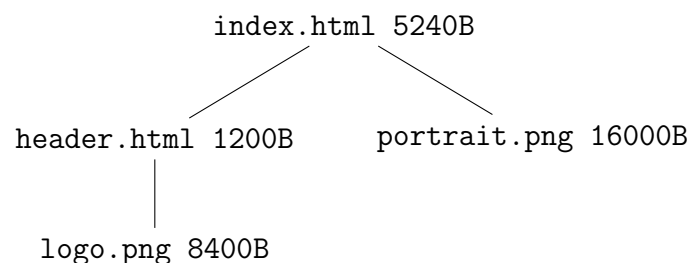
The number next to each file is the size of the file in bytes. For instance, `index.html` is 5240 bytes. Note that `header.html` and `portrait.png` cannot begin to be requested until the transmission of `index.html` is complete. Similarly, `logo.png` cannot begin to be requested until `header.html` is complete.

Based on this, how many seconds would it take for a client to fetch all resources in the figure above? Assume the transmission rate is 800 bits/second from both client-to-server and server-to-client. We will also make some simplifying assumptions to make the calculation easier.

- You can disregard all forms of delay.

- Do not include TCP headers in your calculations.

- Assume only a single TCP connection is used, and that it is already established.

- Each HTML request/response header is 26 bytes.

Question 2: Network Stack Shuffle

After doing an overview of the layers in the network stack, you may have noticed that other choices could have been made in terms of in what layer to put what functionality. Make an argument for moving some functionality up or down the stack (e.g., encryption, re-ordering, routing, etc..). Justify why you think this functionality would be better served at this layer.