# Lab 10 notes:

# Author: Khang Trinh – 102118468

### Part A

Q1: The different types of file open modes are:

Learning Link: <https://www.tutorialspoint.com/cplusplus/cpp_files_streams.htm>

* app: append mode. All output to the file will be appended at the end
* ate: the file will be opened for output and the read/write control will be from the end of the file
* in: open a file for reading
* out: open a file for writing
* trunc: if the file already exists, its contents will be truncated before opening



Fig . I didn't know what "Truncated" means

Q2: Potential reasons for what would happen if you don’t close a file after opening it

Learning Link: <https://stackoverflow.com/questions/15854526/why-is-it-necessary-to-close-a-file-after-using-it>

* Consume unnecessary system resources
* Your changes to the file may not be saved
* Closing the file = releasing resources. Doing it manually = having control over when/how the programmer wants to release the resources

Q3: A char is 1 byte, an int is 4 bytes and a float is 4 bytes, and the file was 9 bytes. It makes sense!

Learning Link: <https://docs.oracle.com/cd/E19253-01/817-6223/chp-typeopexpr-2/index.html>

### Part B

* When reading, it’s worth having “while we’re not yet at the end of the file” for more safety precautions
* At first, I thought using myfile << line << endl; would be good, but then I found a more efficient example to learn from using getline()
* I also found a good example for splitting a string via inputting a delimiter. However, with the current code it only spits out strings, further improvements would need to be made if the output needed to be in other types of data