I created the program that based on input reads number of threads in and out, input file and output file and a log file. It copies character(byte) by character from old file into the new one which it creates or replaces. Also, it makes a log file showing the progress of the work along the way.

1. A clear statement about the assigned work/components you believe you have done/completed successfully.
2. Read input
3. We create n in threads and m out threads
4. Read the whole file into temporary buffer
5. Create threads n in and m out ones
6. IN producer threads get a character and its offset from file (temp buffer that we created) and put it in our main buffer which is a structure of structures which has a char and its offset as data. Also, puts info data into log file along the way.
7. OUT consumer threads get a char and offset from main buffer (one thread – one char at once), and put it into a new created or replaced file in the same position as it was in the original file
8. We use mutex and semaphores to let threads handle their part without other interrupting

So that everything is being done as the tasks requires.

1. we wait for threads depending on the number of IN threads and OUT threads

No thread does everything at once, they do tasks one after another in parallel with synchronization.

1. 1 in and 1 out would not be super fast but let’s say 5 in and 5 out or 50 in and 50 out do job very fast

Text

Description automatically generated

Text

Description automatically generated