**Operating Systems**

**Final Project Part 3**

The final project consists of modifications to the producer-consumer system. For part 3, you will make minor changes to all the components, in preparation for the last 2 parts (part 4 and 5).

In part 2, you changed the order of events, so that rather than placing an item filled with text in the item buffer, a producer places an item which is just an advertisement of some data (size) to be picked up at place (producer socket). When a consumer gets an item from the buffer, it can then consume the item by *stream*ing the advertised number of bytes from the producer client (socket) to the consumer client (socket). This allows the number of bytes to be very large (gigabytes) without exceeding the limitations of any hardware we run the programs on. Make the following changes and then test with large values for MAX\_LETTERS, say 1-2 billion.

Server:

Do not create a huge buffer and read all the data from the producer before sending it to the consumer, use **streaming**, which is basically a “read a little – send a little” model. You wouldn’t like it if Netflix downloaded a whole 2 hour movie before it started playing it, and even if you didn’t mind waiting minutes for a download, some devices don’t even have the space to store whole movies.

Producers:

Do not create the whole set of data before sending it, just *stream* it, keeping whatever model you were previously using to generate random letters.

Consumers:

Rather than *stream* letters into excessively huge text files, stream the letters into a special file called “**/dev/null**”. This **device file** can be written to, but the data is just thrown away. However, keep the code that creates a file named by the thread id, and in that file write either:

* If the number of bytes received is not equal to the number advertised
  + ERROR: bytes (followed by the number of bytes read)
* If the correct number of bytes were received
  + SUCCESS: bytes (followed by the number of bytes read)
* Or if the server closes earlier (before or when trying to get the size)
  + ERROR: REJECTED

These strings are defined in the new prodcon.h