Operation systems, Security and Networks (207SE)
Lab 16: **IPC and Synchronisation II**

Producer/Consumer problems

Your task

In a paragraph explain what the producer/consumer problem is.

The producer/consumer example code provided isn't perfect.

- It's got a fixed-length buffer (which is OK),
- and can only use it once (which is not).

Here are my requirements:

- It should be possible to have an arbitrary buffer length
- It should be possible to push an arbitrary amount of data through the shared buffer
- The data should not be corrupted
- That means: the consumer should not be able to read an empty buffer and the producer should not be able to write to a full one.

For up to 2 marks: Include the description of the producer/consumer problem. Adapt the example code so it has different buffer sizes and that the producer/consumer loops back to the start. The consumer and the producer will not be aware of where each other is and so there is no way to prevent one writing/reading over the other.

For up to 4 marks: Implement a ringbuffer that allows single bytes to be written/read by the producer/consumer. As noted above the producer should not over write data not yet read by the consumer and the consumer should not read data in a slot where the producer has not added new data.

For upto 5 marks: Show the 4 marks code works with different speeds for the consumer and the producer.

Hint: Adapt the prodcon_example2.c code on moodle. You will need the se207_sems.h in the same directory as prodcon_example2.c.

Evidence

Include the description of the producer/consumer problem. The Commented code to complete the task. Output from the working program.