**Assignment 5**

**Due**: 11:55PM October 16, 2023

**Submission**: Complete your submission on Blackboard.

Email submission will NOT be accepted. If you cannot complete the assignment before the deadline due to Covid-19, you must contact the instructor **BEFORE** the deadline. Otherwise, **late submission will NOT be accepted**.

**Total points**: 20 points

1. If you are writing a multi-threaded program, propose a solution to exchange information between two threads in the same process? (5 points) (Hint: Threads in the same process share the memory space).

**Defining global variables in the process could allow two threads in the same process to communicate with each other since they both share the same memory space and therefore have access to the same pointers to those variables.**

2. Assume you are designing a game server. Use **your own words** to design a multithreaded server. (5 points)

**A game server could create a new thread for each user client that connects to the server. This would also allow for the user clients to be run simultaneously if the server is running on a multiprocessor machine. Another way to design a multithreaded server would be to split the world into different portions where each portion is handled by a single thread.**

3. Use your own words to compare the advantages and disadvantages between user-level and kernel-level threads. (10 points)

**User level threads have the advantage of being able to switch between each other without needing the OS to step in. The way they’re scheduled is also entirely dependent on only the developer of the application and they’re more cross platform so they can run on any OS. A disadvantage, however, is that all threads will stop if the process gets blocked. Another disadvantage is that only one user level thread, within a process, can run on a single CPU core at a time.**

**Kernel level threads, within the same process, have the advantage of being able to run simultaneously on different CPU cores. Also, the OS blocks specific threads instead of the whole process, so one kernel level thread getting blocked won’t block the rest of the kernel level threads. A disadvantage, however, is that thread switching must be done by the OS which may result in a slow down.**

END