ELEE 390 B – Developing Concurrent Software – Final Exam

Please answer each question as completely as possible.

**Name:**

1. What is the difference between a process and a thread?

**Threads share memory by default, processes do not. (Threads lecture, slide 3)**

If a std::condition\_variable is signaled, but the predicate passed to the wait method returns false, will the mutex be locked when the wait method exits?

**Yes (Asynchronous operations lecture, slide 9)**

What must you do to get the value of the std::future<T> returned by a call to std::async?

**A call to get waits until the future is ready, then returns the value of type T** **(Asynchronous operations lecture, slide 14)**

Name one of the two relationships that an atomic variable can be used to specify. Briefly explain it.

**(Atomics lecture, slide 11)**

* ***synchronizes-with***
  + **Occurs only with atomics**
  + **Causes a load and a store to work together**
  + **An atomic write (store) synchronizes-with an atomic read(load) on the same memory location**
* ***happens-before***
  + **Not specific to atomics or threads**
  + **If A is sequenced B, A happens-before B**
  + **If A happens-before B, and B is on another thread, then we have an *inter-thread happens-before***
  + **A happens-before B and B happens-before C implies A happens-before C (transitive property)**

Why do we need to use a std::atomic<bool> instead of a bool to signal a background thread to exit?

**A bool does not allow us to tell the compiler about the inter-thread happens-before relationship. (Atomics lecture, slide 14)**