Ch.4 – Threads

- Threads are fundamental unit of CPU utilization that forms the basis of multi-threaded computer systems.
- Process creation is heavy-weight while thread creation is light-weight.
  - Can simplify code and increase efficiency.
- Kernels are generally multi-threaded.
- Multi-threading models include: Many-to-One, One-to-One, Many-to-Many.
  - Many-to-One: Many user-level threads mapped to single kernel thread.
  - One-to-One: Each user-level thread maps to kernel thread.
  - Many-to-Many: Many user-level threads mapped to many kernel threads.
- Thread library provides programmer with API for creating and managing threads.
- Issues include: thread cancellation, signal handling (synchronous/asynchronous), handling thread-specific data, and scheduler activations.
  - Cancellation:
    - Asynchronous cancellation terminates the target thread immediately.
    - Deferred cancellation allows the target thread to periodically check if it should be canceled.
  - Signal handler processes signals generated by a particular event, delivered to a process, handled.
  - Scheduler activations provide upcalls – a communication mechanism from the kernel to the thread library.
    - Allows application to maintain the correct number of kernel threads