Threads

Practice Exercises

4.1 Provide two programming examples in which multithreading provides better performance than a single-threaded solution.

4.2 What are two differences between user-level threads and kernel-level threads? Under what circumstances is one type better than the other?

4.3 Describe the actions taken by a kernel to context switch between kernel-level threads.

4.4 What resources are used when a thread is created? How do they differ from those used when a process is created?

4.5 Assume an operating system maps user-level threads to the kernel using the many-to-many model and the mapping is done through LWPs. Furthermore, the system allows developers to create real-time threads. Is it necessary to bind a real-time thread to an LWP? Explain.

4.6 A Pthread program that performs the summation function was provided in Section 4.3.1. Rewrite this program in Java.