

Lab 4 -- Java Threads

1. Communication and Synchronization

1. Build and implement the Bank program without using **synchronized** methods. Check the results for interference. Add the correct synchronization. Check that the interference disappears.
2. Write a concurrent Java program which declares an array of 10 integers. Create two threads. One which writes the value 1 to all the elements in the array. However, in between writing each element of the array, the thread sleeps for one second. The second thread writes the values 7 to each element of the array. Again in between writing the elements it sleeps. When both threads have terminated, print out the array. Understand what is happening.
3. Now modify the program so that, even with the sleeps, the result at the end of the threads is (1,1,1,1,1,1,1,1) or (7,7,7,7,7,7,7,7,7,7). You will need to use a synchronized method or a synchronized statement to achieve this.
4. Work through this Sun tutorial on [Synchronizing threads](#). Build and implement the examples. Make sure that you understand them. Make small modifications to each program to test your understanding.
5. Work through this Sun tutorial on [High Level Concurrency Objects](#). Provide a new implementation of the **TenThreads** example that uses a thread pool. Use a fixed size pool. Try your program with a variety of pool sizes, e.g. <10, =10 and >10. Observe and explain any differences in behaviour.