

## Homework 2

1. [2] What are *asynchronous*, *deferred*, and *disabled* cancellation in pthreads? What makes them different?
2. What is the role of the `contentionscope` for pthreads? How does it affect the execution of pthreads?
3. Assume you have a system that does not provide a `usleep(unsigned long usec)` call to suspend the execution of the thread for a given amount of time, say in  $\mu$ secs. How would you implement this function using condition variables? (Describe your implementation in form of a C/C++ code snippet.)
4. [1] Which of the scheduling algorithms listed below could result in starvation?
  - (a) First-come, first-served
  - (b) Shortest job first
  - (c) Round robin
  - (d) Fixed priority
5. [1] Servers can be designed to limit the number of open connections. For example, a server may wish to have only  $N$  connections active at any time. As soon as  $N$  connections are established, the server will not handle more connections until at last one existing connection is closed. How would you use semaphores to limit the number of concurrent connections? Assume that you have a function called `HandleNextIncomingConnection()` and one called `CloseExistingConnection()`, which are called before and at the end of handling a connection. How would you add the necessary synchronization code to limit the number of concurrent connections to at most  $N$ ?
6. [3] Ten processes share a critical section implemented by using a semaphore  $x$ . Nine of these processes use the code `x.P(); <critical section>; x.V()`. However, one process erroneously uses the code `x.V(); <critical section>; x.P()`. What is the maximum number of processes that can be in the critical section at the same time?
7. [3] A customer gives the following instructions to a bank manager: Do not credit any funds to my account if the balance in my account exceeds  $n$ , and hold any debits until the balance in the account is large enough to permit the debit. Design a class in Java (called `SafeAccount`, to be initialized with a given value for  $n$ ) with two methods, `credit` and `debit`, that implements this type of account.

## References

- [1] A. Silberschatz, P. Galvin, and G. Gagne, *Applied Operating Systems Concepts*, John Wiley & Sons, Inc., New York, NY, 2000.
- [2] Deitel, Deitel, and Choffnes, *Operating Systems*, Pearson / Prentice Hall, 2004.
- [3] D. M. Dhamdhere, *Operating Systems, a Concept-Based Approach*, McGraw Hill, 2009.