Technische Universität München *Summer Semester 08*
Fakultät für Informatik *K N Verma*
Prof. Dr. H. Seidl *verma@in.tum.de*

Virtual Machines

Exercise Sheet 12

Deadline: 15 July 2008, during lecture, by email, or in room 02.07.041

Exercise 1: *6 Points*

Use the *pthreads* library to implement the semaphore type `Sema` and the operations `newSema`, `Up` and `Down` discussed in the lecture (without using predefined semaphore functions). You will need functions like `pthread_mutex_init`, `pthread_mutex_lock`, `pthread_mutex_unlock`, `pthread_cond_init`, `pthread_cond_wait`, `pthread_cond_signal`.

Exercise 2: *14 Points*

The *dining philosophers problem* consists of N philosophers seated around a table. One chopstick is placed between each pair of philosophers. Each philosopher spends some amount of time thinking, after which he gets hungry and wants to eat. To eat, a philosopher needs to pick up the chopsticks on his left as well as on his right. After eating he puts down both chopsticks and restarts thinking until he is hungry again, and continues like this forever. The problem is to devise a protocol so that every hungry philosopher eventually gets both pairs of chopsticks for eating. Implement each philosopher as a thread. Use printing commands to display the state of the philosophers and chopsticks from time to time.