Prof. Dr. Oscar Nierstrasz
Pascal Gadient, Leonel Merino

Series 01 — 20.09.2017 – v1.0
Introduction to Concurrency

Exercise 1 (7 Points)

Answer the following questions (1 point each):

1. Do recent central processing units (CPUs) of desktop PCs support concurrency? Why became concurrency for many software applications very important these days?
2. What is safety? Give one concrete example of a safety violation.
3. What is liveness? Give a concrete example of a liveness violation.
4. Using the implementation in the slides, can a binary semaphore lead to a deadlock? Can it lead to starvation? Explain with the aid of an example.
5. Why do we need synchronization mechanisms in concurrent programs?
6. How do monitors differ from semaphores? Please provide a precise answer.
7. How are monitors and message passing similar? And how are they different?

Exercise 2 (2 points)

```
x := 1
Thread 1 -> x := x + 7.
Thread 2 -> x := x * 5.
```

Considering the previous code, give all possible values of x at the end of the execution of both threads with their corresponding execution traces.

Hint: You should be able to perceive 6 different execution flows in total, however, some of them could may lead to the same x.

Exercise 3 (1 points)

Implement a monitor using semaphores. Use pseudo-code and comment it.