

# Concurrent Programming

## Practical 4: The Sleeping Tutor

The aim of this practical is to model the following scenario using monitors and/or semaphores.

A tutor is sleeping in his room, waiting for two students to arrive for a tutorial.<sup>1</sup> The first student to arrive sleeps while waiting for her tutorial partner. The second to arrive wakes the first, and one of them wakes the tutor. The two students then sleep while the tutor gives the tutorial. At the end of the tutorial, the two students wake up and leave. The tutor sleeps until the next tutorial.

For simplicity, we will assume that there is just a single pair of students, but they may return for multiple tutorials.

Outline code to capture this scenario is given below, and is available from the course website. (Note that in this code, “sleep” corresponds to a thread sleeping, not the tutor or a student sleeping!)

```
val random = new scala.util.Random;

def Student(me: String) = proc("Student"+me){
  while(true){
    sleep(random.nextInt(2000));
    println("Student_"+me+" arrives"); Arrive;
    println("Student_"+me+" ready for tutorial"); ReceiveTute;
    println("Student_"+me+" leaves");
  }
}

def Tutor = proc("Tutor"){
  while(true){
    println("Tutor_waiting_for_students"); TutorWait;
    println("Tutor_starts_to_teach"); sleep(1000);
    println("Tutor_ends_tutorial"); EndTeach; sleep(1000);
  }
}

def System = Tutor || Student("Alice") || Student("Bob")
```

---

<sup>1</sup>We suspect that many students believe their tutor to be like the light inside a refrigerator – inasmuch as it turns on only when the door is opened. We regret that the scenario outlined here may add verisimilitude to this otherwise unfounded belief.

Your task is to implement the following four procedures:

**TutorWait** The tutor waits for the students to arrive;

**Arrive** A student arrives;

**ReceiveTute** A student receives a tutorial;

**EndTeach** The tutor finished the tutorial.

The two requirements are:

- The tutor starts to teach only after both students have arrived;
- The students leave only after the tutor ends the tutorial.

You should implement these procedures either

- using a monitor, or
- using semaphores.

**Optional:** do both.

Your report should be in the form of a well commented program, describing any design decisions you have made.

Gavin Lowe and Bernard Sufrin  
5<sup>th</sup> January, 2012(3.573)