Universitetet i Oslo
Institutt for Informatikk
PSY (ConSeRNS)
Einar Broch Johnsen, Silvia Lizeth Tapia Tarifa,
Chinmayi Baramshetru(group teacher)

uiologo

INF 5170: Models of Concurrency

Fall 2021 — **Mandatory assignment 2** — 6. Oct. 2020

Issued: 6. Oct. 2020

Due: 1. Nov. 2020

1 General remarks

How to deliver

- Your solution should be delivered online (https://devilry.ifi.uio.no/).
- Program examples should be commented in order to make them understandable for the group teacher or lecturer.

Who delivers

We encourage to work together in groups of 3 (but not more). For "technical" reasons (devilry): in a group each member should upload the same solution (which should be identical, just the same PDF uploaded several times).[1] The solution must be marked with names and email addresses of all contributing students.

Check in time that devilry works and that your status within devilry (and student web etc) is ok. Don't try your devilry access as late as the day of the deadline. In case of doubts, for clarifications, or if having trouble with devilry etc, ask in time.

Evaluation

This assignment is graded *pass* or *fail*.
You must pass the obligs in order to take the final exam.

[1] That facilitates managing acceptance.

The Roller Coaster problem

Exercise 5.17 (a) in Andrews. A number of *passengers* (1:n) wish to repeatedly take rides with a *Roller Coaster*. There should be one process for each passengers and one for the Roller Coaster. As for the problem of the *sleeping barber*, the processes must participate in several synchronization stages. The passengers must wait for the car to be present, and the car must wait for `C` passengers to enter. After the ride, the car must awake all riding passengers. The solution does not have to prevent sneaking. Make a solution in the setting of *monitors with signal and continue discipline* for signaling.

In addition, try to formulate a reasonable invariant based on your solution, but it is *not* necessary to prove the invariant using programming logic.

Greatest common divisor

Let n and m be positive integers. The following program will find the greatest common divisor $\gcd(n, m)$ of n and m:

```
1
2   x := n, y := m;          // "input" m and n
3
4   while (x ≠ y) {
5
6   co
7
8     < if (x > y) x:= x−y >
9
10  ||
11
12    < if (y > x) y:= y−x >
13
14  oc
15  }                        // the value of x should now be gcd(n, m)
```

You may assume the following properties about the greatest common divisor (gcd):

$$\begin{aligned} &x > y \Rightarrow \gcd(x, y) = \gcd(x - y, y) \\ &\gcd(x, x) = x \\ &\gcd(x, y) = \gcd(y, x) \end{aligned} \tag{1}$$

Do the following:

1. Prove that the parallel program, if it terminates, has done what it is supposed to do, i.e. use *programming logic* to prove the following *post condition*

   $$\{\ x = \gcd(n, m)\ \}\ . \tag{2}$$

   Use

   $$\gcd(x, y) = \gcd(n, m) \tag{3}$$

   as *loop invariant*.

2. The program above consists of a `while` loop repeatedly executing an inner `co`-statement. This means that two processes are started and terminated each time the loop body is executed. Is it possible to develop a corresponding program with a `co` statement at the outermost level and a while loop in each of the processes? The program should have the same postcondition. Explain your answer. What about interference?