ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

D. Basin and P. Müller

Formal Methods and Functional Programming

Exercise Sheet 12: Modeling

Submission deadline: May 26th, 2009

Assignment 1

In the lecture, you have seen parts of the modeling language *Promela*. In this and later assignments you will use Promela and the model checker *Spin* for modeling and analyzing concurrent programs. You can find detailed information about how to install the model checker Spin on your computer at the webpage `http://spinroot.com/spin/Man/README.html`.

Some useful information for running Spin:

- If Spin is invoked without any options, it performs a *random simulation*. For example,

  ```
  $ spin foo.pr
  ```

  performs a random simulation for the Promela model specified in the file `foo.pr`.

- The command line option `-a` generates a protocol *specific analyzer*. The output is written into a C file pan.c, which you can, e.g., compile with the GNU C compiler. Running the compiled file will explore the state space of the Promela model and check whether the model might deadlock. The assertions in the model are also checked.

  ```
  $ spin -a foo.pr
  $ gcc -o pan pan.c
  $ ./pan
  ```

Note that there is also a graphical interface for Spin, called XSpin, which you can also download from the Spin webpage `http://spinroot.com`. We recommend that you also install XSpin on your computer. See `http://spinroot.com/spin/Man/Manual.html#S` for more information.

The purpose of this assignment is to get familiar with Promela and the model checker Spin.

(a) Consider again the following statement (which you already saw on previous exercise sheets).

```
y := 0;
while x > 0 do
  y := y + x;
  x := x - 2
end
```

Write a model in Promela to check if the statement starts in a state σ with $\sigma(\mathtt{x}) = 3$, it will reach a state σ' with $\sigma'(\mathtt{y}) = 4$.

(b) Write a model in Promela to verify that the following program could result in a state in which x has the value 1 or 4.

$$x := 1 \,[]\, x := 2; x := x + 2$$

(c) Write a model in Promela to verify that the following program could result in a state in which x has the value 1 or 3 or 4.

$$x := 1 \texttt{ par } x := 2; x := x + 2$$

(d) Consider the following program.

```
x := 5;
y := 1;
(while x > 1 and y < 5 do
   (x := x - y [] y := y + 1)
 end
par
 while x > 0 do
   y = y + 1;
   x = x - 1
 end)
```

Assume that we start the program in some state. Can we reach a terminal configuration in which the variable x stores the integer -7? What is the minimal value of the variable x after executing the program?

Assignment 2

The problem of the dining philosophers illustrates common issues for concurrent programs. N philosophers sit around a circular table. Each philosopher spends her/his life thinking and eating. In the center of the table is a large platter of spaghetti. Because the spaghetti are long and tangled a philosopher must use two forks to eat them. Since the philosophers can only afford N forks, a single fork is placed between each pair of philosophers, which they have to share. Each philosopher can only reach the forks to her/his immediate left and right with her/his left and right hand, respectively.

(a) You find a skeleton (`philosopher_skeleton.pr`) of a Promela model of the dining philosophers at the course webpage. Complete the model such that each philosopher chooses nondeterminstically to pick up her/his left or right fork first. Use Spin to find a deadlock and explain its source. How does the falsification time change when increasing the number N of philosophers.

(b) The deadlock can be avoided when each philosopher behaves more deterministically to pick up the forks (not all philosophers have to behave the same). Model your solution in Promela and use Spin to check that it is indeed deadlock free. Is your model also starvation free (i.e., each philosopher will eventually eat)?

Assignment 3

In the lecture, you have seen a Promela program that models parts of the Needham-Schroeder protocol, namely agent Alice and an intruder. Complete the Promela program for agent Bob according to the description of the protocol. Use the template from the course webpage. Simulate your model in (X)Spin. Can you find an attack?