Final Exam, Reactive Systems 2000

Tuesday, March 14, 08.00 - 14.00. Polacksbacken.

For assistance, you may bring and consult any notes, course material, books, etc. that you like. In Swedish:

Alla hjälpmedel tillåtna

This exam consists of 5 problems. The problems are formulated in English. Start solution of each problem on a new sheet of paper. Write your name on each sheet. You may write in English or in Swedish, whichever you prefer.

Before starting to work on any problem, first read through the entire exam.

Problem 1 (10 p) The following is an action system with two integer variables x and y. It does not perform too many useful things.

$$
\begin{array}{l}
\textbf{ActionSystem} \;\; \mathit{Foo1} \\
\quad \textbf{variables} \\
\qquad x, y \; : \; \text{integer} \\
\quad \textbf{initially} \\
\qquad x = y = 0 \\
\quad \textbf{actions} \\
\qquad x \leq y \;\longrightarrow\; x \;:=\; x + 1 \\
\quad [\!] \\
\qquad x \geq y \;\longrightarrow\; y \;:=\; y + 1 \\
\quad [\!] \\
\qquad x > y + 1 \;\longrightarrow\; y \;:=\; y + 10 \\
\textbf{end}
\end{array}
$$

For each of the three actions, there is a (weak) fairness constraint, defined by letting each action be a (weak) fairness set. Which of the following properties are satisfied by the program? Motivate your answers:

a) $\Box\ (x \geq y)$

b) $x = 0\ \ leads-to\ \ x = 10$

c) $x > y\ \mathcal{U}\ x < y$

d) $x \geq y\ \mathcal{U}\ x < y$

e) $x \geq y\ \mathcal{W}\ x < y$

f) $(x > y + 1)\ \mathcal{W}\ (y = x + 20)$

Problem 2 (6 p) The following is an action system with two integer variables x and y. It does not perform too many useful things.

$$
\begin{array}{l}
\textbf{ActionSystem} \;\; \mathit{Foo2} \\
\quad \textbf{variables} \\
\qquad x, y, z \; : \; \text{integer} \\
\quad \textbf{initially} \\
\qquad x = y = z = 0 \\
\quad \textbf{actions} \\
\qquad x \leq y \;\longrightarrow\; x \;:=\; x + 1 \\
\quad [\!] \\
\qquad x \geq y \;\longrightarrow\; y \;:=\; y + 1 \\
\quad [\!] \\
\qquad x > y \;\longrightarrow\; z \;:=\; z + 1 \\
\quad [\!] \\
\qquad x < y \;\longrightarrow\; z \;:=\; z + 1 \\
\textbf{end}
\end{array}
$$

Under which of the following assumptions about fairness does this action system satisfy the property

$$z = 0\ \ leads-to\ \ z = 10$$

a) Each of the four actions is a weak fairness set.

b) Each of the four actions is a strong fairness set.

c) There are two weak fairness sets, one containing the first two actions, and one containing the last two actions

d) There are two strong fairness sets, one containing the first two actions, and one containing the last two actions

Problem 3 (10 p) Mr. Promel wanted to model and analyze a protocol in SPIN. He Produced a Promela model with two channels and two processes, which contained the following declarations and statements.

```
mtype = { m0, m1, ack};

proctype Sender
{
        (* Code for sender *)
}

proctype Receiver
{
        (* Code for Receiver *)
}

init
{       chan StoR = [5] of { mtype };
        chan RtoS = [5] of { mtype };
        atomic {
                StoR!m0;        /* start */
                run Sender;
                run Receiver
        }
}
```

He found that the model was too big (too many reachable states). He then wanted to make a more abstract model, where instead of representing the state of a channel by a sequence of messages, the model only remembers *how many* messages are in the channel. He then replaced the above declarations by

```
mtype = { m0, m1, ack} ;
byte    StoR, RtoS ;

proctype Sender
{
        (* Modified Code for sender *)
}

proctype Receiver
{
        (* Modified Code for Receiver *)
}

init
{       atomic {
                StoR = 1 ;      /* start */
                run Sender;
                run Receiver
        }
}
```

In the code for the Sender and for the Received, he modified send and receive statements as follows:

- A send statements of form `StoR!m0` is replaced by `StoR = StoR + 1`
- A receive statement of form `StoR?m0` is replaced by
  `StoR = (StoR > 0 -> (StoR - 1) :  StoR)`
- Analogously for send and receive statements to other channels and with other messages.

Problem: What is wrong with the above abstraction? How should it be done properly?

Problem 4 (8 p) Assume that we model a system which interacts with the environment via variables *req*, *conf*, and *warning*. We wish to verify the following property:

At each state where *req* is true, it is either the case that in some following state either

- the variable *conf* is true, or
- the variable *warning* becomes true and stays true for the entire remainder of the computation.

Your problem is

a) to describe how to express this property in temporal logic,

b) Assume that we have a PROMELA model of the system. What additional code do you add in order to use SPIN to validate that the above property holds for the system.

Problem 5. (6 p) Prove that there is no *deterministic* Büchi Automaton, which recognizes exactly the set of infinite sequences that satisfy $\Diamond \Box p$, where p is a state formula.

Hint: Assume that such a deterministic automaton exists. Consider how it behaves when it sees a prefix of some word that satisfies $\Diamond \Box p$: eventually, e.g., after seeing a long sequence of p-states, it must get to an accepting state. After the automaton gets to an accepting state, assume that it sees a sequence of states satisfying $\neg p$: eventually it must get to a non-accepting state. Then we feed it states satisfying p, etc. Thus, the automaton can be made to oscillate between accepting and non-accepting states. Based on observations like this, you should be able to prove that a deterministic Büchi automaton can not accept exactly the sequences in $\Diamond \Box p$.