Prof. Bernd Finkbeiner, Ph.D.
Peter Faymonville, M.Sc.
Michael Gerke, B.Sc.

Winter term 2011/2012
Problem Set 2

Verification

Please write the names of all group members on the solutions you hand in.

Problem 1: Model Checking with SPIN

Program TRY-MUX1 of Figure 1 is again suggested as a tentative solution to the mutual exclusion problem. For this exercise (and for Problem 2), we will use the Model Checker SPIN (`http://spinroot.com`) to automatically verify some of last week's properties. For now, use assertions appropriately to verify the presence of path or the absence of a path in the system.

$$\textbf{local } y_1, y_2 : \textbf{integer where } \; y_1 = 0, y_2 = 0$$

$$P_1 :: \left[\begin{array}{l} l_0 : \textbf{loop forever do} \\ \left[\begin{array}{l} l_1 : \textbf{noncritical} \\ l_2 : \textbf{wait until } y_2 = 0 \\ l_3 : y_1 := 1 \\ l_4 : \textbf{critical} \\ l_5 : y_1 := 0 \end{array} \right] \end{array} \right] \; || \; P_2 :: \left[\begin{array}{l} m_0 : \textbf{loop forever do} \\ \left[\begin{array}{l} m_1 : \textbf{noncritical} \\ m_2 : \textbf{wait until } y_1 = 0 \\ m_3 : y_2 := 1 \\ m_4 : \textbf{critical} \\ m_5 : y_2 := 0 \end{array} \right] \end{array} \right]$$

Figure 1: Program TRY-MUX1: proposed solution.

(a) Implement TRY-MUX1 in Promela. Check basic functionality using simulation runs and printf-statements.

(b) Verify that both critical regions are accessible, i.e. that there exist paths to the critical regions of P1 and P2.

(c) Verify the mutual exclusion property.

(d) Answer questions (b) and (c) for a modified version of the program, TRY-MUX2, in which statements l_2 and l_3 are interchanged and so are statements m_2 and m_3.

Please send your Promela-files of TRY-MUX1 and TRY-MUX2 to `mailto:faymonville@cs.uni-saarland.de`. Be prepared to demo your verification runs in your next discussion slot, either on your own laptop or by sending us all necessary files.

Problem 2: Communication channels with SPIN

There is an easy solution to the mutual exclusion problem if one can communicate via channels instead of using shared memory.

(a) Develop a Promela model of a mutual exclusion protocol for two processes, uses just one channel and no shared variables.

(b) Use SPIN to check wether your protocol satisfies mutual exclusion.

(c) Does your protocol avoid starvation? Give an informal argument why it does or a human readable counter example why it doesn't.

(d) Assume you have four processes and a shared ressource that can be accessed by two processes at once, say a quadcore processor where only two cores at a time can read the RAM. Adopt your model from (a) to this scenario.

Please send your Promela-files of (b) and (d) to `mailto:faymonville@cs.uni-saarland.de`. Be prepared to demo your verification runs in your next discussion slot, either on your own laptop or by sending us all necessary files.

Problem 3: Fairness

Consider the transition system TS shown in Figure 2 with the set of atomic propositions $\{a\}$.

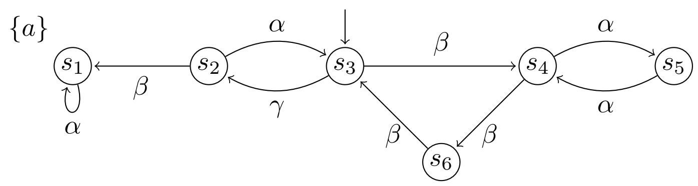

Figure 2: Transition system TS

Let the fairness assumption

$$\mathcal{F} = (\emptyset, \{\{\alpha\}, \{\beta\}\}, \{\{\beta\}\})$$

determine whether $TS \vDash_{\mathcal{F}}$ "eventually a". Justify your answer!