University of Innsbruck Institute of Computer Science
Exam 28 January 2008

Introduction to Model Checking (VO) WS 2007/2008 LVA 703503

First name: ____________________

Last name: ____________________

Matriculation number: ____________________

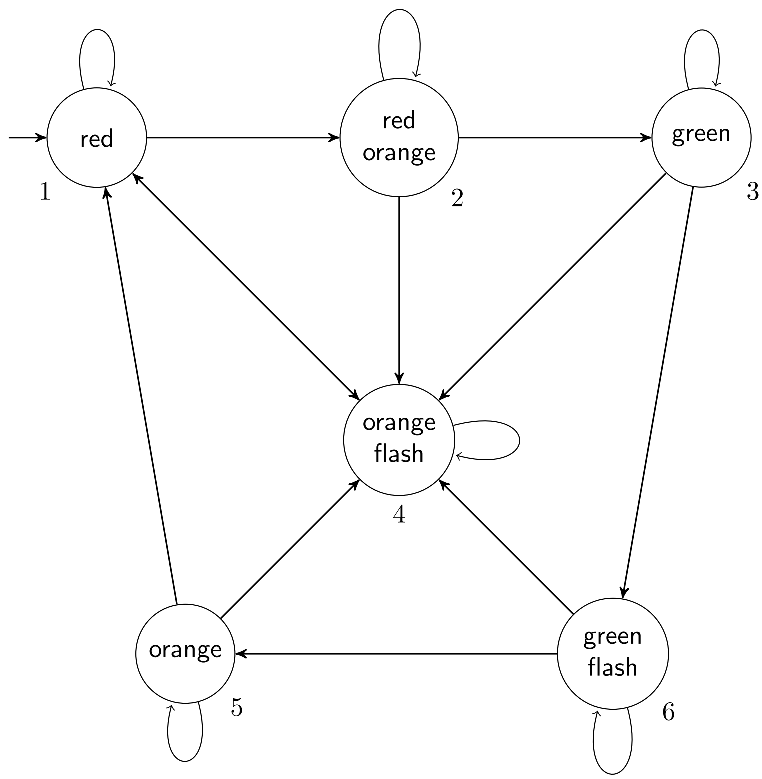
- Write your name and matriculation number on every page.
- Please answer all exercises in a readable and precise way. Do *not* write with a pencil or a red pen. Please cross out solution attempts which are replaced by another solution.
- Cheating is not allowed. Everyone who is caught will fail the exam.
- Please do not remove the staples of the exam.

Exercise	**Maximal points**	**Points**
1(i)	**12**	
1(ii)	**5**	
2	**20**	
3	**15**	
4	**18**	
Σ	**70**	
Grade		

First name	Last name	Matriculation number

Exercise 1 (12 + 5 points)

Consider the following transition system TS of an Austrian traffic light using the atomic propositions $\{\mathsf{red}, \mathsf{orange}, \mathsf{green}, \mathsf{flash}\}$.



Here, state 4 represents a traffic light which is out-of-order.

(i) Consider the following property.

It is never allowed to switch from a red to a non-red state
if the red state is not additionally orange. $(\star)$

One might formulate this property as the following CTL*-formula Φ.

$$\Phi = \neg\mathsf{red} \vee \mathsf{A}\,(\mathsf{X}\,\mathsf{red})\,\mathsf{U}\,(\mathsf{red} \wedge \mathsf{orange})$$

Compute $Sat(\Phi)$ by applying the CTL*-model checking algorithm. Here, the sets $Sat(\Psi)$ should be indicated for every state-subformula Ψ of Φ. Note that the subformula $\mathsf{red} \wedge \mathsf{orange}$ of Φ should be seen as a state-formula.

It is not necessary to perform the LTL-model checking explicitly, but write down the LTL-formula that is checked.

- $Sat(\mathsf{red}) = \{1, 2\}$
- $Sat(\neg\,\mathsf{red}) = \{3, 4, 5, 6\}$
- $Sat(\mathsf{orange}) = \{2, 4, 5\}$
- $Sat(\mathsf{red} \wedge \mathsf{orange}) = \{2\}$

First name	Last name	Matriculation number

- $Sat(\mathsf{A}\,(\mathsf{X}\,\mathsf{red})\,\mathsf{U}\,(\mathsf{red} \wedge \mathsf{orange}) = \{2\}$ (This steps involves LTL model checking of the formula $\mathsf{X}\,\mathsf{red}\,\mathsf{U}\,a$ where a is a new atomic proposition representing the state-formula $\mathsf{red} \wedge \mathsf{orange}$. Hence, only state 2 is labeled with a.)
- $Sat(\Phi) = \{2, 3, 4, 5, 6\}$

(ii) By the results of part (i) one can easily see that $TS \not\models \Phi$. There are two reasons for this.

- Φ does not correspond to property $(\star)$.
- Property $(\star)$ does not take care of the possibility that the traffic light can become out-of-order.

Specify two formulas in the logic of your choice (LTL, CTL, CTL*) to fix these problems.

- One formula should exactly formalize $(\star)$.
- The other formula should integrate the out-of-order possibility.

- $(\star)$ can be encoded in LTL as follows:

$$\mathsf{G}\,((\mathsf{red} \wedge \mathsf{X}\,\neg\mathsf{red}) \Rightarrow \mathsf{orange})$$

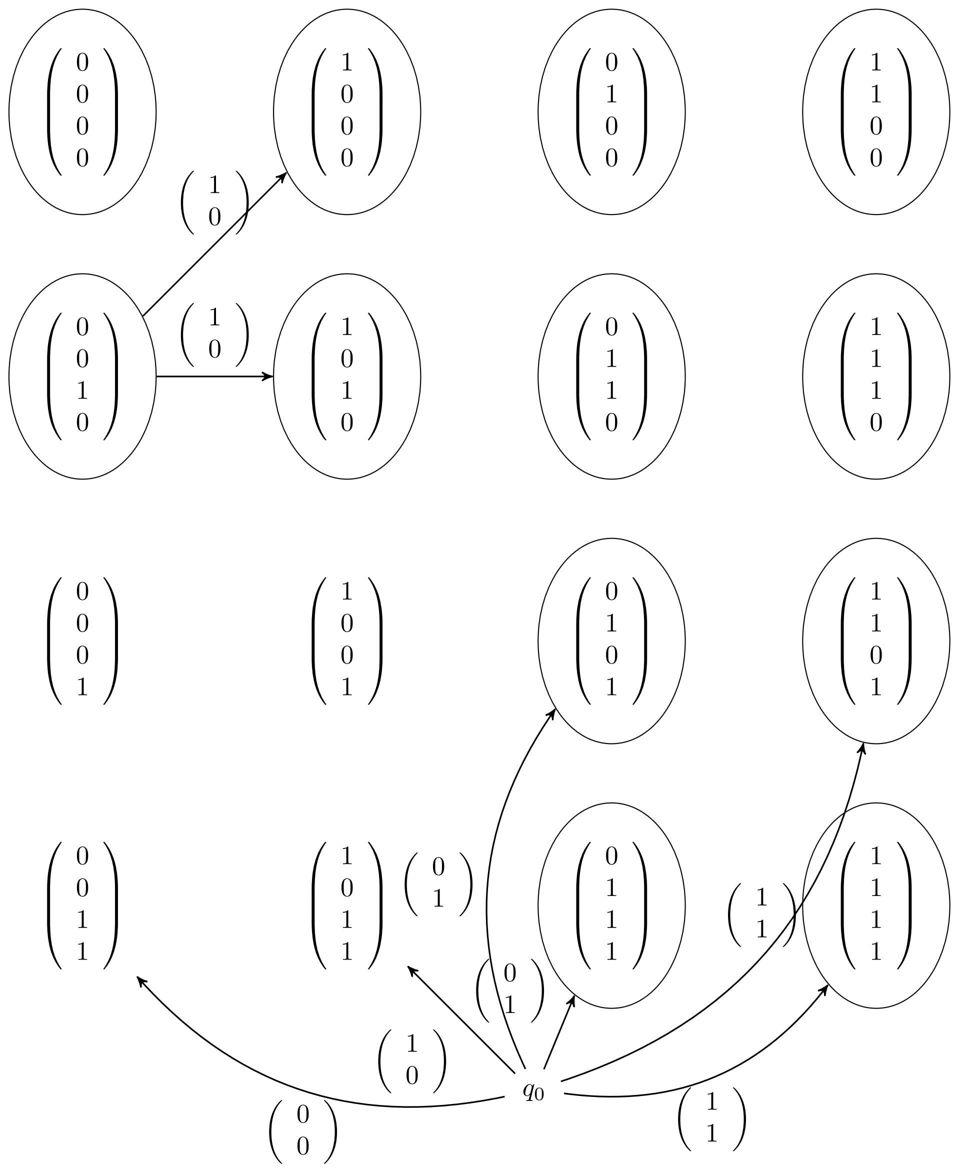
- Taking care of out-of-order can be done as follows:

$$\mathsf{G}\,((\mathsf{red} \wedge \mathsf{X}\,\neg\mathsf{red}) \Rightarrow (\mathsf{orange} \vee \mathsf{X}\,(\mathsf{orange} \wedge \mathsf{flash})))$$

First name	Last name	Matriculation number

Exercise 2 (20 points)

Consider the LTL-formula $\varphi = \mathsf{X}\, a \,\mathsf{U}\, b$ where a and b are atomic propositions. Using the construction to generate a GNBA from φ we obtain an NBA with the following states.



- Draw a circle around every final state.
- Draw all outgoing transitions of $\begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$ and all outgoing transitions of q_0.

Here, the vectors represent the formulas $\begin{pmatrix} a \\ b \\ \mathsf{X}\, a \\ \mathsf{X}\, a \,\mathsf{U}\, b \end{pmatrix}$.

First name	Last name	Matriculation number

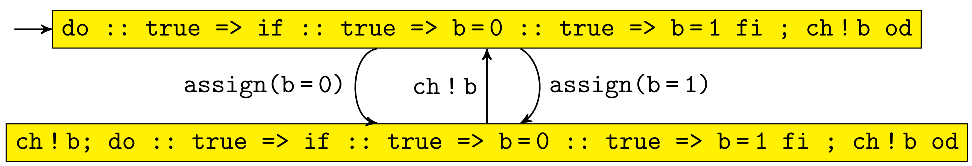
Exercise 3 (15 points)

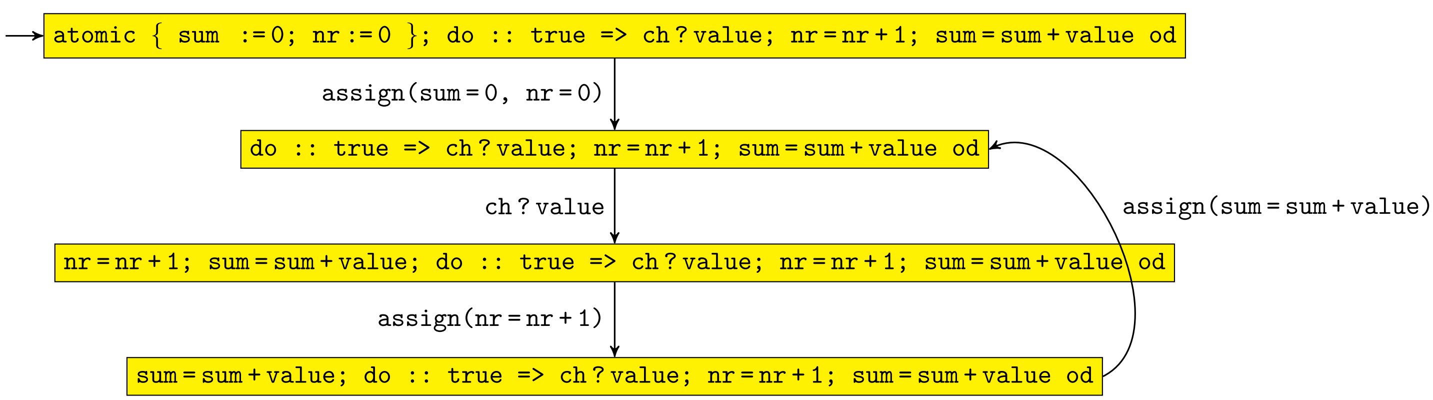
Construct the channel-system for the following nanoPromela program which models a sender which sends bits to a receiver via a lossy channel. The receiver computes the sum of the bits and the number of bits transferred.

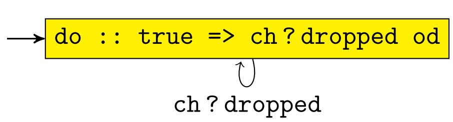
```
------- SENDER PROCESS ------------
do :: true => if :: true => b = 0 :: true => b = 1 fi ; ch ! b od

------- RECEIVER PROCESS ----------
atomic { sum := 0; nr := 0 };
do :: true => ch ? value; nr = nr + 1; sum = sum + value od

------- LOSSY CHANNEL PROCESS -----
do :: true => ch ? dropped od
```







First name	Last name	Matriculation number

Exercise 4 (18 points)

Each correct answer is worth two points. A wrong answer withdraws one point. Marking both "Yes" and "No" is a wrong answer only and does not result in $2-1=1$ points. It is not possible that the total score of this exercise is negative.

	Yes	No
The CTL formula $\mathsf{A}\,\mathsf{F}\,\mathsf{A}\,\mathsf{F}\,b$ is equivalent to the LTL formula $a\,\mathsf{U}\,\mathsf{F}\,b$. ($\mathsf{A}\,\mathsf{F}\,\mathsf{A}\,\mathsf{F}\,b \equiv \mathsf{A}\,\mathsf{F}\,b \equiv \mathsf{F}\,b \equiv a\,\mathsf{U}\,\mathsf{F}\,b$)	✓	
There is an LTL formula which is equivalent to $\neg\mathsf{E}\,a\,\mathsf{U}\,\mathsf{X}\,b$. ($\neg\mathsf{E}\,a\,\mathsf{U}\,\mathsf{X}\,b \equiv \neg\neg\mathsf{A}\,\neg(a\,\mathsf{U}\,\mathsf{X}\,b) \equiv \mathsf{A}\,\neg(a\,\mathsf{U}\,\mathsf{X}\,b) \equiv \neg(a\,\mathsf{U}\,\mathsf{X}\,b)$)	✓	
The size of the transition system of a program graph is polynomial in the number of locations of the program graph. (Yes, since the states of $\mathit{TS}(\mathit{PG})$ are $\mathit{Loc} \times \mathit{Eval}(\mathit{Var})$.)	✓	
There exists a transition system, a state s, and a CTL*-state-formula Φ such that neither $s \models \Phi$ nor $s \models \neg\Phi$. (No, see definition of CTL*-semantic: $s \not\models \Phi$ implies $s \models \neg\Phi$.)		✓
Assume $P \neq NP$. The Hamiltonian Path Problem can be encoded in polynomial time as a CTL model checking problem. (No. If a polynomial time encoding would exist, then one can solve HPP in polynomial time, since CTL model checking can be done in polynomial time. This is a contradiction to HPP being NP-complete.)		✓
There exists an NBA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{A}) \neq \mathcal{L}(\varphi)$ for every every LTL formula φ. (Yes, e.g. the NBA on slide 8 of lecture 10.)	✓	
Let channel c use hand-shaking. Then no state with an empty channel c can perform a transition which involves receiving a value along c. To be more precise, there is no state $\langle \ell_1, \ldots, \ell_i, \ldots, \ell_n, \eta, \xi \rangle$ where $\ell_i \xrightarrow{c?x}_i \ell'_i$ is a transition in the program graph and where $\xi(c) = \varepsilon$ such that a transition to a state of the form $\langle \ell'_1, \ldots, \ell'_i, \ldots, \ell'_n, \eta', \xi' \rangle$ with $\ell_i \neq \ell'_i$ is possible. (Since for handshaking the channels are always empty, of course a receiving process can be executed if there is a suitable second process which sends along c.)		✓
Given a DBA $\mathcal{A}_1$ and a GNBA $\mathcal{A}_2$, then $\overline{\mathcal{L}(\mathcal{A}_1)} \setminus \mathcal{L}(\mathcal{A}_2)$ is recognizable by an NBA. (DBAs and GNBAs can be transformed into NBAs. Since NBAs are closed under all Boolean operations, one can construct such an automaton.)	✓	
There is an N**F**A $\mathcal{A}_1$ and a GN**B**A $\mathcal{A}_2$ such that $\mathcal{L}(\mathcal{A}_1) = \mathcal{L}(\mathcal{A}_2)$. (Choose NFA and GNBA which accept the empty language.)	✓	