NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING
EXAMINATION FOR

SEMESTER 1 2009/2010

CS 3234 - LOGIC AND FORMAL SYSTEMS

December 2009 **Time Allowed: 2 Hours**

INSTRUCTIONS TO CANDIDATES

This is an open book examination. Any written and printed material may be used during the examination. The examination consists of a paper part, and an automated theorem proving part.

Paper Part (54 marks)

1. Enter your matriculation number here:

2. Enter the answers to the questions in this answer book in the provided spaces.
3. This examination booklet has 12 pages, including this cover page, and contains 8 questions.
4. Answer all questions.
5. The maximal marks achievable is indicated for each question.

Automated Theorem Proving Part (36 marks)

1. find the questions in the file `final.v` on the desktop of the computer in front of you.
2. **Enter your name and matriculation number in the designated section on top of the file.**
3. Read and follow the instructions in the file.
4. Save the file after entering the answers; do not move the file.

Do not write below this line

Q#	1	2	3	4	5	6	7	8	A	B	C	D	Σ
Max	5	6	8	3	10	6	11	5	9	9	9	9	90
Sc													

Predicate Calculus

Question 1: (5 marks) Prove the following sequent in predicate calculus:

$$\vdash \forall x((\forall y P(x,y)) \rightarrow P(x,x))$$

Question 2: (6 marks) Prove that the following formula in predicate calculus is not valid:

$$\begin{aligned}\forall x \forall y \exists z_1 \exists z_2 (\quad & z_1 \neq z_2 \wedge \\ & P(x, z_1) \;\wedge\; P(x, z_2) \;\wedge \\ & P(y, z_1) \;\wedge\; P(y, z_2) \;\wedge \\ & \rightarrow x = y)\end{aligned}$$

Question 3: (8 marks) Consider the following grammar of a restricted predicate logic syntax:

$$\begin{array}{rcl} t & ::= & x \mid f(t_1, \ldots, t_n) \\ \phi & ::= & P(t_1, \ldots, t_m) \mid (\neg\phi) \mid (\phi \wedge \phi) \mid (\phi \vee \phi) \mid \\ & & (\phi \rightarrow \phi) \mid (\forall x\ \phi) \mid (\exists x\ \phi) \mid \end{array}$$

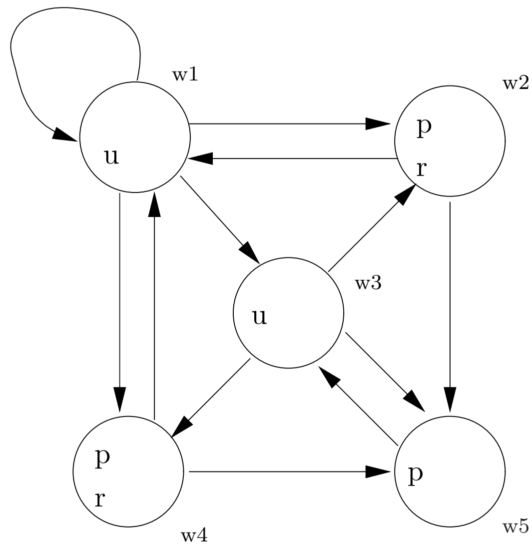
where $n > 0$ and $m > 0$. The restriction excludes constants (nullary function symbols).
A variable occurrence in a formula is a place where a variable x appears as term (not right after $\forall$ or $\exists$), and a predicate occurrence in a formula is a place where a predicate symbol P appears. For example, the formula

$$(\forall x(P(x, y) \wedge \exists z(P(y) \vee Q(x, y, z))))$$

has 6 variable occurrences and 3 predicate occurrences.
Prove that every formula ϕ in this language has at least as many variable occurrences as it has predicate occurrences.

Modal Logic

Question 4: (3 marks) Consider the Kripke model depicted in the following diagram.



List all worlds x for which $x \Vdash \Diamond\Box p$ holds. (no proof required)

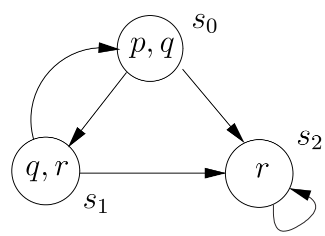
Linear Time Logic

Question 5: (10 marks) Consider the following additional construct for LTL:

$$\phi \ ::= \ \ldots \mid (F_{[n,m]}\ \phi)$$

where $n \geq 0$, $m \geq 0$ and $\pi \models (F_{[n,m]}\ \phi)$ iff there is some i with $n \leq i \leq m$ such that $\pi^i \models \phi$.

- (2 marks) Consider the following model $\mathcal{M}$:



Does $\mathcal{M}, s_0 \models F_{[2,3]}q$ hold? Explain your answer in one sentence.

- (2 marks) Translate the formula $F_{[1,1]}p$ to an equivalent LTL formula without this new construct.

item (2 marks) Translate the formula $F_{[1,2]}p$ to an equivalent LTL formula without this new construct.

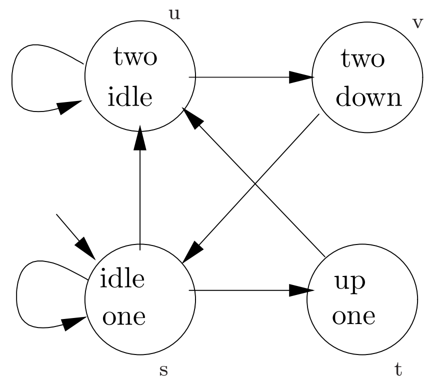
- (2 marks) Translate the formula $F_{[5,9]}p$ to an equivalent LTL formula without this new construct.

- (2 marks) Translate the formula $F_{[7,4]}p$ to an equivalent LTL formula without this new construct.

- (4 marks) Complete the following translation function *translate* such that it can be used to automatically translate formulas from the described extended form to an equivalent LTL formula without the new construct.

$$
\begin{array}{rcl}
translate(\top) & ::= & \top \\
translate(\bot) & ::= & \bot \\
translate(p) & ::= & p \\
translate(\neg\phi) & ::= & \neg translate(\phi) \\
translate(\phi_1 \wedge \phi_2) & ::= & translate(\phi_1) \wedge translate(\phi_2) \\
translate(\phi_1 \vee \phi_2) & ::= & translate(\phi_1) \vee translate(\phi_2) \\
translate(\phi_1 \rightarrow \phi_2) & ::= & translate(\phi_1) \rightarrow translate(\phi_2) \\
translate(X\ \phi) & ::= & X\ translate(\phi) \\
translate(F\ \phi) & ::= & F\ translate(\phi) \\
translate(G\ \phi) & ::= & G\ translate(\phi) \\
translate(\phi_1\ U\ \phi_2) & ::= & translate(\phi_1)\ U\ translate(\phi_2) \\
translate(\phi_1\ W\ \phi_2) & ::= & translate(\phi_1)\ W\ translate(\phi_2) \\
translate(\phi_1\ R\ \phi_2) & ::= & translate(\phi_1)\ R\ translate(\phi_2) \\
translate(F_{[n,m]}\ \phi) & ::= & \ldots
\end{array}
$$

Question 6: Consider the following LTL model for a lift that goes between two levels.



The proposition *up* indicates that the lift is going up, the proposition *down* indicates that the lift is going down, the proposition *idle* indicates that the lift is idle (not moving), the proposition *one* indicates that the lift is at Level 1, and the proposition *two* indicates that the lift is at Level 2.

- (2 marks) Formulate the following requirement in LTL and state (without proof), whether the model $\mathcal{M}$ at starting state s_0 satisfies the requirement:

  If the lift is at Level 2 and going down, then the lift must reach Level 1 at the next step.

- (2 marks) Formulate the following requirement in LTL and state (without proof), whether the model $\mathcal{M}$ at starting state s_0 satisfies the requirement:

  If the lift is idle, it remains idle until it either goes up or down.

- (2 marks) Formulate the following requirement in LTL and state (without proof), whether the model $\mathcal{M}$ at starting state s_0 satisfies the requirement:

  If the lift is at Level 1, it must go up in order to reach Level 2.

Question 7: (12 marks) Consider the following program P, written in the Core language of the lectures.

```
z = 0;
b = 0;
x = 0;
while (z <> y) {
   if (b = 0) {
     b = 1
   } else {
     b = 0;
     x = x + 1
   };
   z = z + 1
}
```

Prove the following Hoare triple:

$$\vdash_{par} (\!|\exists n(n + n = y)|\!)\ P\ (\!|x + x = y|\!)$$

Question 8: (5 marks) Consider the following proposal for an additional proof rule for Hoare logic:

$$\frac{(\!|\phi|\!)\ P\ (\!|\psi|\!)}{(\!|\phi \wedge \eta|\!)\ P\ (\!|\psi \wedge \eta|\!)}\text{[Frame]}$$

Is this a sound rule? Explain your answer.

END OF QUESTIONS