Prof. Dr. Heike Wehrheim
Jürgen König

Paderborn, October 12th 2018
Due date: October 22th 2018

Model Checking

WS 2018

Assignment 1

Submission

Assignments must be submitted via PANDA.

!! Due date is Monday, 22th of October !!

EXERCISE 1 (10 Points) (First Order Logic) **:**
Reiterate the concepts and rules of first order logic like quantifiers and predicates. Then solve the following exercises:

a) Express the following natural sentences in first order logic using the predicates visits(x, y), wears(x, y)

- Jenny wears a suit.
- Jenny visits a bar.
- If Jenny visits a construction site, Jenny wears a helmet.
- Whenever Alex visits a LARP convention, he wears chain mail.

b) Let the universe be $\mathbb{U} = \mathbb{N}$. All mathematical operators have their usual meaning. List for each formula if it is valid, satisfiable and which of the other formulas are a logical consequence of it.

- $\forall\, x \exists\, a(x \cdot 3 = a)$
- $\forall\, x(x \cdot 3 = a)$
- $\forall\, x(x \cdot y > x^2 \Rightarrow y > x)$
- $5 \cdot 3 = a$
- $\exists\, a(x = a + 1)$

EXERCISE 2 (Spin) **:**
Try to install Spin as specified in the Spin Install Guide. SPIN will be needed in this assignment. The easiest way is using Linux or a VM running Linux.

EXERCISE 3 (6 Points) (Kripke Structures) **:**
Recall the definition of a Kripke structures. Then solve the following exercises:

a) Let $(\{s_0, s_1, s_2, s_3, s_4\}, \{s_0\}, \{(s_1, s_2), (s_2, s_3), (s_2, s_4), (s_4, s_1), (s_0, s_1), (s_0, s_3), (s_3, s_3)\}, L)$ be a Kripke structure where $L(s_i) = CR_i$, $0 \leq i \leq 4$, draw this Kripke structure.

b) Give 2 paths of this structure.

c) Take a look at Figure 1, write down the Kripke structure in tuple form.

d) Give a path for this Kripke structure.

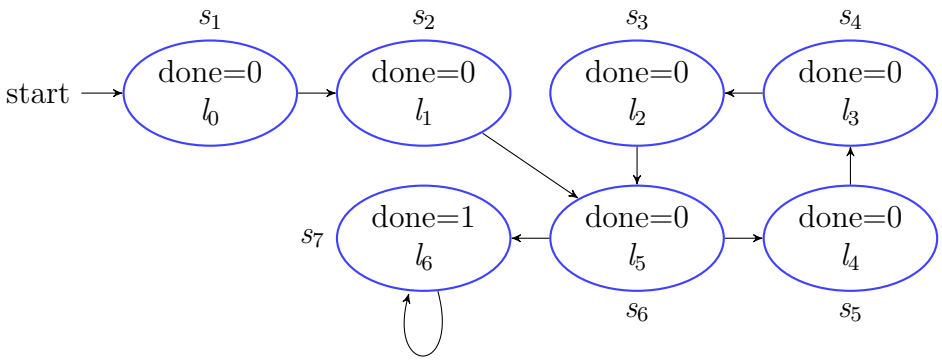

Figure 1: A Kripke structure

EXERCISE 4 (14 Points) (Promela) **:**
Recall the properties and structure of Promela. Then solve the following exercises:

a) Give the possibly values (implicitly) of y at the end of the execution for all possible interleavings.

```
int x = 0;
int y = 0;
int z = 0;
byte b = 0;

proctype One(){
x = 3;
z = 2;
b = 1;
}

proctype Two(){
do
:: (x == 3) -> y = y + 1;
:: (z == 0) -> y = 5;
:: (b == 1) -> break;
od
}

init { run One(); run Two() }
```

b) Consider the dining philosophers problem:

- Three silent philosophers sit around a circular table. On the table is a bowl of rice.
- Between each pair of adjacent philopophers is one chopstick.
- Each philosopher either eats or thinks.
- In order to eat, a philosopher must acquire both chopsticks. When finished eating, a philosopher puts the chopsticks back.

Implement the problem in Promela, modeling each philosopher as a process. Use the greedy solution to the problem, in which each philosophers tries to take available chopsticks whenever he is hungry.
This solution must be submitted in a text file.

c) Run an interactive simulation in Spin. Try to find a run that leads to no philosopher getting to eat, attach a screenshot of that run.