Technische Universität München
Institut für Informatik
Prof. Dr. Helmut Veith
Dipl.-Inf. Michael Tautschnig

SS 2007
Übungsblatt 1
Mittwoch, 9. Mai 2007

Übung zur Vorlesung Model Checking

SMV

Use NuSMV to solve the following exercises. NuSMV can be obtained from `http://nusmv.irst.itc.it`

(a) Write an SMV program that implements an 8 bit comparator. Given two binary strings $a_1 a_2 \cdots a_8$ and $b_1 b_2 \cdots b_8$, the program outputs 1 if $(a_1 a_2 \cdots a_8)_2 \leq (b_1 b_2 \cdots b_8)_2$, where $(s)_2$ denotes the number that results from interpreting s in binary notation, and zero otherwise. Simulate the program in the interactive SMV environment.

(b) Write an SMV program that implements an 8 bit adder and simulate the program in the interactive SMV environment.

(c) *Token ring.* A token ring consists of m independent processors which are arranged in a cycle, where each processor is connected to its left and right neighbors. The processes of the token ring use a token (represented as a Boolean flag) to synchronize each other. After each processing step, the token is passed on to one of its neighbors.

- Implement the token ring for $m = 4$, where the token is passed to one of the neighbors nondeterministically. Simulate the token ring in the interactive environment.

- Use SMV to check whether it is guaranteed that each processor gets access to the token infinitely often. Check the same specification for the case where the processors pass on the token only to its right neighbors.

- In both setups, show that no two processors have access to the token at the same time.

- Suppose now that there are two tokens in the token ring; each processor can hold up to two tokens simultaneously. After each processing step, the processor passes all its tokens nondeterministically to the neighbors. Answer the following questions: Does every processor get at least one token infinitely often? Does each processor get hold of both tokens simultaneously infinitely often? Evaluate these questions also in the case one token is always passed to the right neighbor whereas the second token is always passed to the left one.

(d) * *Dining Philosophers (due to Dijkstra, 1968).* Five philosophers sit around a circular table. One fork is placed between each pair of philosophers; one fork will be shared between exactly two philosophers. In the center of the table there is a large

plate full with spaghetti. A philosopher spends his life either eating spaghetti or thinking. To eat, the philosophers need to pick up *both* forks that are immediately right and left to their plates separately (unfortunately, both forks cannot be picked up concurrently). Philosophers are very patient—they might hold a fork to the end of times.

Model the dining philosophers problem in SMV. We want to assure that no philosopher starves to death. More precisely, check whether each philosopher eats infinitely often in any possible future.

(e) * Use SMV to find a path from A to B in the following labyrinth (it is possible to move horizontally and vertically):