Tik-79.146 Logic in Computer Science: Special Topics I **Spring 1998**
Homework assignment 3
20.4.1998

Deadline 11.5.1998.

The home assignment can be returned at the lecture or to the box by the room Y418.

CTL Model Checking

The aim of this assignment is to specify and model check some properties of a small mutual exclusion system. The system has been created with the `PROD` Pr/T -Net reachability analysis tool and will be checked with the model checker built into `PROD`. All the necessary software has been installed to the computing center HP workstations (the "tree" -series of workstations).

The modeled system is a three process mutual exclusion system using the ticket algorithm.

The algorithm has a pair of atomically accessed global variables *next* and *granted*. They are modeled by a token in place *next_ granted*. In model checking you can use atomic propositions of the form: `(next_granted >= <.2,1.>)`, which is true in all states in which next has value 2 and granted has value one.

The program counters are modeled by 3 tuples in the place `pc`. The tuples are of the form `<.pid, state.>` where $0 \leq pid \leq 2$ and $state \in \{rem, enter, test, leave\_try, crit, reset, leave\_exit\}$. In model checking you can use atomic propositions of the form: `(pc >= <.2,crit.>)`, which is true in all states in which process 2 is in the critical section. Example: `check ef (pc >= <.2,crit.>)`.

The local variables ticket are modeled by 3 tuples in the place `vars`. The tuples are of the form `<.pid, value.>` where $0 \leq pid \leq 2$ and $0 \leq value \leq 3$, where 3 is a null value. In model checking you can use atomic propositions of the form: `(vars >= <.0,0.>)`, which is true in all states in which process 0 has ticket value 0.

The assignment is to specify the following properties with CTL and present the model checking results for the formulas in the initial state of the system:

1. Any two processes are never at the same time in the critical section.
2. When process 0 gets in to the enter section, it will always eventually get to the critical section.
3. The system can from all system states return to a state in which all processes are in the remainder section.
4. Process 0 can from all system states eventually reach the critical section.
5. In all system states if the process 2 is in the critical section then it will always eventually get into the remainder section.

Did any of the properties have unexpected outcomes? Which ones did and why do you think they did?

Please see the course web-page for detailed instructions on how to use the model checker. Include in your answer the properties above specified in CTL, and answers to the questions above. Also include output of a CTL model checking session checking all the properties. (You can obtain this for example by using the "script" -command.)