CS 720 Spring 2003
Take-Home Final

Due: May 23, 5 PM
Do not discuss this test with anyone else except me.

Please sign the statement below and include this sheet when you hand in the test.

I have not given or received any help on this test which is not allowed. I have not discussed this test with anyone else.

Signature: ____

CS 720 Spring 2003
Take-Home Final

1. Consider the LTL formula $AG\neg(CR_0 \wedge CR_1)$ and the Kripke structure of Figure 2.2 of CGP.

   (a) Using the method of Section 9.4 of CGP, translate the negated path formula $\neg G\neg(CR_0 \wedge CR_1)$ into a Büchi automaton $\mathcal{S}$. Show all steps of your work, as we did in class, and then draw the final automaton.

   (b) Using the method of Section 9.2 of CGP, convert the Kripke structure of Figure 2.2 into a Büchi automaton $\mathcal{B}$.

   (c) Using the method of Section 9.2, give a Büchi automaton $\mathcal{A}$ for $\mathcal{L}(\mathcal{B}) \cap \mathcal{L}(\mathcal{S})$. (Use the method that takes advantage of the fact that every state of $\mathcal{B}$ is accepting.) Show only the states that can be reached from initial states. Do not give every state of $\mathcal{B}$.

   (d) Explain why $\mathcal{L}(\mathcal{A}) = \emptyset$. (If you did the previous steps correctly, this will be obvious. You will not have to apply the double DFS algorithm.)

[25 points]

2. (a) For each transition α in the transition system in Figure 10.9 of CGP, give the smallest possible value of $pre(\alpha)$, ie, those transitions which must be in $pre(\alpha)$ according to the definition of pre.

   (b) For each transition α of the transition system, give the set $pre(\alpha)$ you get if you use the heuristics in Section 10.5.2 of CGP.
   [Hint: For at least one transition, the pre you get using the heuristics is larger than the minimum possible.]

   (c) If you use the function *ample* on page 159 of CGP, what sets do you get for $ample(\alpha)$ when you do partial order reduction on the structure in Figure 10.9? List your answers only for those states that you reach in your depth first search. Explain your answers in detail. If a set $T_i(s)$ does not meet all the tests, state which one it fails. If it passes all the tests, explain in particular why it passes the test $check\_C1$. Always consider process 0 before process 1. Do you get the same reduced state graph as in Figure 10.9?

[25 points]

3. Section 6.4 gives a complicated way of finding witnesses with fairness constraints. If there are no fairness constraints, witness finding becomes much easier.

   (a) Suppose that $f = \mathbf{E}[f_1 \mathbf{U} f_2]$ and that you have the sequence of approximations $Q_0, Q_1, \ldots, Q_i$ that converge to the set of states where f is true as described on page 68. Give an algorithm which, given a state s in which f is true, determines a path starting at s which makes f true. The path will be finite and lead to a state where f_2 is true, with f_1 true along the way.

(b) Suppose that $f = \mathbf{EG} f_1$ and you have the sequence of approximations that converges to the set of states where f is true, as given on page 68. Describe an algorithm that finds an infinite path starting at s in which f_1 is true at all points on the path.

(c) Describe how you can implement the previous two algorithms using OBDDs. Assume that the approximations Q_i are given to you as OBDDs.

[25 points]

4. Theorem 2 is the main result of the paper "On the expressive power of CTL combinators" that you looked at for Homework 2. The proof is in the paper, but is somewhat sketchy. Give a more complete proof, keeping in mind the following:

   - The proof in the paper uses an infinite Kripke structure. Our Kripke structures have to be finite. You will have to replace the infinite structure with a series of finite ones and give a claim similar to the ones we gave when we proved that the different logics have different expressive powers.
   - The proof given in the paper of Lemma 3 is a good outline, but there are some details for you to fill in. Also, there is at least one small problem in the argument given that you will have to fix up.

[10 points]

5. (a) Download SMV from the URL given on page 215 of HR. There is a manual at this webpage, but you can learn all you need from Section 3.6 of HR and Section 8.1 of CGP.

   (b) Give an SMV program that simulates the Kripke structure of Figure 2.2 as closely as you can. (You can use the program of Section 3.6.3 of HR as your starting point.) Give a printout of your program.

   (c) Translate the four properties on pages 170-171 of HR into CTL formulas that are appropriate for your program. Add them as SPECs and give a printout of the result.

   (d) Add a FAIRNESS condition that makes more of your SPECs true.

   (e) Is there any of your SPECs that you cannot make true no matter what FAIRNESS constraint you add? Explain.

[15 points]