Homework 4
Due Date: Friday, November 29, 2002

Somesh Jha
Computer Sciences Department,
University of Wisconsin,
Madison, WI 53706.

Question 1 (20 points):
(Part a:) Express the relations shown in Figure 1 using the relations given in class and various operators (of course!). Please give a short justification for your answer.
(Part b:) Let `CousinsFatherSide`(n), `CousinsMotherSide`(n), and `Cousins`(n) be the set of cousins that are separated by n-steps on the father's side, mother's side, and father or mother's side. Write a recursive expression for these relations, i.e., express the relations with parameter n in terms of relations with parameter $n-1$ and the basic relations introduced in class.

Question 2 (20 points): Establish whether the following equations be-

`FatherInLaw` : `People` $\leftrightarrow$ `Males`
`SisterInLaw` : `People` $\leftrightarrow$ `Females`
`FirstCousinsFatherSide` : `People` $\leftrightarrow$ `People`
`FirstCousinsMotherSide` : `People` $\leftrightarrow$ `People`
`FirstCousins` : `People` $\leftrightarrow$ `People`
`SecondCousinsFatherSide` : `People` $\leftrightarrow$ `People`
`SecondCousinsMotherSide` : `People` $\leftrightarrow$ `People`
`SecondCousins` : `People` $\leftrightarrow$ `People`

Figure 1: Some relations

tween the relations are true or not. Please justify your answer. You should enter these equations in *Alloy* and play around with them. Assume that the relations have appropriate types.

$$(P;Q)^\top = Q^\top;P^\top$$
$$(P \subseteq Q) \Rightarrow (P;P \subseteq Q)$$
$$(Q \neq \emptyset) \Rightarrow (\texttt{Un};Q;\texttt{Un} = \texttt{Un})$$

`Note:` The symbol $\emptyset$ denotes the empty relation.

Question 3 (20 points): Translate the phone-switch example we covered in class into a model accepted by Alloy. Use Alloy to check the assertions and explain the counterexamples you obtain. Now propose a fix and incorporate it in the model.

Question 4 (40 points): In this question, you will write a specification for the problem described. Our aim is to model a library. There are two types `PERSONS` and `BOOKS`. There is a function `issued` : `BOOKS` $\rightarrow$ `PERSONS` and a set `Library` : *Set* `BOOKS`. If book b is issued to a person p, then `issued`$(b) = p$. `Library` represents the set of books in the library. There is also a set of books on reserve given by the set `Reserve` : *Set* `BOOKS`. The operations are:

- *Issue a book*
  This operation issues a book to a person. A book on reserve or already issued cannot be issued.

- *Return a book*
  This operation models the act of a person returning a book.

- *Adding a book to the library*
  This operations models a new book being added to the library.

- *Putting a book on reserve*
  This operations models a book being put on reserve. A book which is currently issued cannot be put on reserve.

- *Taking a book off reserve*
  This operation models a book being taken off reserve.

The claims (or assertions in Alloy) are:

- Issuing a book and then returning it results in the same state.
- If a book is on reserve, it should never be issued. This claim should be true before and after every operation.

(Part a:) Write a mathematical description for all the operations and claims and explain your answer.
(Part b:) Now express your design in *Alloy*.