

CS 42, Fall 2012
Assignment 9
Sequential Machines and Prolog Part 1
Due Wed. Dec. 5, 11:59 PM

1. Design a finite-state acceptor that expects a sequence of 0's and 1's as input and that accepts exactly those sequences that represent a number divisible by 5 in binary, most-significant bit first. For example, your machine accepts any sequence of all 0's (including the empty sequence, by convention), 101, 1010, 1111, 10100, 11001, 11110, ..., which represent 5, 10, 15, 20, 25, 30, respectively, but does not accept 1, 10, 11, 100, 1100, 1101, ..., which do not represent multiples of 5.

Your design should be similar to the machine in the lecture slides that accepts multiples of 3, but will have more states.

Submit your design as a state-transition diagram on paper, not through the submission site.

On the same paper, give the next-state equations for the following implementation.

The next two parts should be submitted as separate circuits in a single Logisim file **a09.circ**.

2. Construct, using Logisim, a clocked sequential logic implementation of your machine using a 1-hot encoding for the states.
3. Construct, using Logisim, a clocked sequential logic implementation using a binary encoding of the states.

The remaining parts should be submitted as a single Prolog file, **a09.pro**.

4. Construct a Prolog predicate **m5** that expects a list of 0's and 1's as an argument and which succeeds in cases when the acceptor designed in part 1 accepts.
5. Construct the following Prolog predicates that work with the movies database, `movies.pro`, which is posted on the CS 42 website. There is also a skeleton file **a09.pro**.

```
% ans0 is solved for you as an example.
% ans0 is true for the director of ['American Beauty', 1999]?
ans0(Director) :- movie(['American Beauty', 1999], Director, _).

% ans1 is true for any movie [Title, Year] in which Harrison Ford played a role.
ans1(_Movie) :- tbd.
```

```
% ans2 is true for any movie [Title, Year] having comedy as a descriptor.
ans2(_Movie) :- tbd.

% ans3 is true for the title of two movies with the same title but produced in
% different years.
ans3(_Title) :- tbd.

% ans4 is true for an actress or actor born in New Jersey
ans4(_Person) :- tbd.

% ans5 is true for an actress or actor born before 1950.
ans5(_Person) :- tbd.

% ans6 is true for an actress or actor born in New Jersey before 1950.
ans6(_Person) :- tbd.

% ans7 is true for a pair of an actor and actress who played together in the same movie.
ans7(_Actor, _Actress) :- tbd.

% ans8 is true for a pair of an actor and actress who played together in more than
% one movie.
ans8(_Actor, _Actress) :- tbd.

% ans9 is true for an actor or actress who both directed and played in the same movie.
ans9(_Person) :- tbd.

% ans10 lists actors and actresses together in order of increasing birth year, giving
% the year first.
ans10(_List) :- tbd.

% ans11 is true for the youngest actress or actor (give multiple results if there is
% a tie).
ans11(_Person) :- tbd.

% ans12(Degree, Person1, Person2) is true if Person1 is within Degree degrees of Person2.
% within degree 1 means that Person1 and Person2 acted in the same movie.
% within degree N > 1 means either Person1 is within degree N-1 of Person2, or
% Person1 acted in a movie with some Person3, and Person2 is within degree N-1 of
% Person3.
% Degree is assumed to be instantiated. It is preferable that each instance of Person2 be
% given only once for an instance of Person1, and vice-versa.
ans12(_N, _Person1, _Person2) :- tbd.

% ans13(N) means that N is the number of actors and actresses not within 6
% degrees of Kevin Bacon.

ans13(_N) :- tbd.
```