Fall 1998
## Midterm #1
Professor Aiken

Problem #1: Regular Expressions and Finite Automata

a. In class we showed how to convert an NFA recognizing the language A Into an NFA recognizing the language A*. Given a machine recognizing A, the following construction adds epsilon transitions in both directions between the machine's start and final states (you may assume there is exactly one final state; there is always one start state in any case). Show that this construction is not a machine for A*. Your answer should be a DFA (not an NFA) accepting a language A that when modified according the construction does not recognize exactly A*.
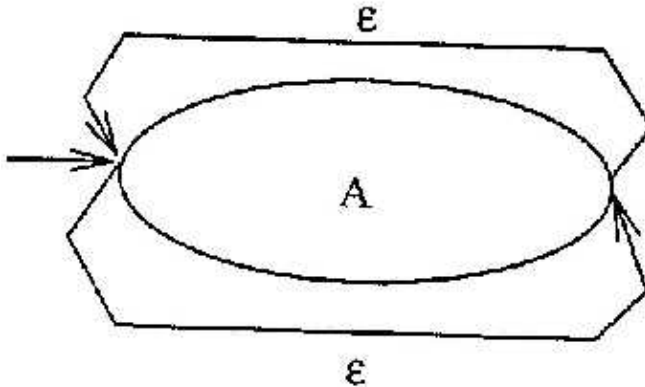
b. Give a regular expression for the set of all strings over 0 and 1 containing a pair of consecutive 0's and a pair of consecutive 1's.

c. Give a DFA recognizing the set of all strings over 0 and 1 containing a pair of consecutive 1's followed-not necessarily immediately-by a pair of consecutive 0's.

Problem #2: First and Follow Sets

a. Some languages are read right-to-left rather than left-to-right (e.g., Hebrew). Right-to-left parsers do not use First and Follow sets, but rather Last and Precedes sets. Give precise (and concise) definitions of Last and Precedes analogous to the definitions of First and Follow given in lecture.

b. Give a grammar with the following two properties:

   ♦ The grammar generates an infinite language (i.e., the language has an infinite number of strings).
   ♦ For every non-terminal A in the grammar, $First(A) = Last(A)$ and $Follows(A) = Precedes(A)$.

Problem #3: Bottom-Up Parsing
Give the action and goto tables of an SLR(1) parser for the following grammar:

```
S -> aSbb
S -> Y
```

```
Y -> c
```

Problem #4: Miscellaneous Grammars

    a. Give a context-free grammar **G** where every string of terminals of length n derivable from G has $2n$ derivations. Your grammer should generate an infinite number of strings of terminals.

    b. Let $R=o^*(S+ epsilon)$. Give a grammar for which R is the set of viable prefixes. S is a non-terminal and 0 is a terminal.

    c. Give a grammar that is not LR(1).