Homework Assignment #1

Due: Friday, January 12, 2006, In Class

0. Write a regular expression for the language of nonnegative integer constants in C, where numbers beginning with 0 are octal constants and other numbers are decimal.
1. Convert the following regular expression to a NFA:
   a((**bcd**|**a***cd**)**x**)*|**x***a**
2.
   a. Convert the following regular expression (where the alphabet is a, b, and c) into an NFA, following the mechanical rules developed in class.
      `(a|b)(a|b)* | (a|b)(a|b)* c (a|b)(a|b)*`
   b. Convert this NFA into a DFA, following the algorithm from class. Be sure to label the NFA states and to label each of the DFA states with a set of NFA states.
3.
   a. The regular grammar specifying lexically correct programs for MiniJava is given as follows: **Program** ::= (**Token**|**Whitespace**)* Modify this specification to require that all tokens be separated by whitespace, and optionally allow whitespace at the start and/or end of the program.
   b. Why does this language-change remove the need for the longest-match meta-rule?
   c. Do you think this would be a good language design change? Justify.

Produce a hard-copy of your answers and turn them in by the start of class on the due date. Do these exercises individually, not with your project partner.