Assignment 1

1. Suppose the language recognized by DFAs **M** and **N** are **L1** and **L2** respectively. How can we use DFAs **M** and **N** to construct a DFA that recognizes the language **L1∩L2**? [Can you use De Morgan's Law?]

2. Show by giving an example that, if **M** is an NFA that recognizes language **L**, swapping the final and non-final states in **M** doesn't necessarily yield a new NFA that recognizes the complement of **L**. Is the class of languages recognized by NFAs closed under complement? Explain your answer.

3. Show that the following grammar is ambiguous, by showing two parse trees representing the string "**id + id * id**". Here, '**+**', '**-**', '*****', '**(**', '**)**', '**id**' are terminals. Also give the two leftmost derivations, and the two rightmost derivations corresponding to the parse trees.

   Expr → Expr + Expr | Expr + Expr | Expr * Expr | (Expr) | id

4. Following is an example of different ways one can declare variables for some programming language. Write down the grammar that can generate such variable declarations. Assume all variable names are one character long, and the variable types are integer, real, char, and boolean.

   var i : integer;

   var b : boolean;

   var m : real;

   c : char;

   x, y, z : integer;

5. Write down a regular definition for floating point numbers in Scientific notation. First write a regular definition for signed numbers and number with decimal point and then use these regular definition to write the regular definition for scientific notation.

Language	Example	Pattern
Signed	+6, -12	All sequences of digits starting with + or −
Decimal	10.541, 45.02	Includes a decimal point and at least one digit to the left and right of the point.
Scientific	-10e-01, - 43E+10	**Signed** or **Decimal** followed by **e** or **E** and a **Signed** number