Given the Prolog interpreter for language ONE, add the modulo operator % to the language (that is, add the modulo operator to the grammar for ONE). The precedence for % is the same as multiply and it is left associative. Extend the Prolog interpreter for ONE with rules for the modulo operation. Use the binary term ‘mod’ as your abstract syntax representation. Demonstrate that your semantic definition computes correct values for modulo. In particular, demonstrate that your semantics computes the correct values for the following terms:
   1. 4%2 should be the value 0
   2. 3*2%3 should be the value 0
   3. 3*(2%3) should be the value 6

Hand in your extended grammar and the source code for the Prolog interpreter. In addition, hand in a screen shot etc to show that your interpreter works.

**Hint:** You will need to transform the terms in the above problems into abstract syntax trees first. The abstract syntax tree for problem 1 above should read `mod(const(4),const(2))`.

**Hint:** Prolog has a built-in operation called ‘mod’ that computes the modulus.