Lab Session Software Testing 2013, Week 3 With each deliverable, indicate the time spent.

1. (If you need it:) Refresh your knowledge of First Order Logic, by consulting `http://www.logicinaction.org/docs/ch4.pdf`.

2. Study the techniques in the *Techniques* slides of today until you have mastered them.

3. Consult the course slides of this week to write a generator for random integer lists. The type should be

   ```
   genIntList :: IO [Int]
   ```

4. A permutation of a finite list is another finite list with the same elements, but possibly in a different order. For example, [0,2,0] is a permutation of [0,0,2], but [2,2,0] is not. Write a function

   ```
   isPermutation :: Eq a => [a] -> [a] -> Bool
   ```

   that returns `True` if its arguments are permutations of each other.

5. Define some testable properties for this function, and use your random generator for integer lists from Exercise 3 to test `isPermutation`.

6. Use the random formula generator from the Techniques slides to test your CNF program of last week. (Deliverable: file with tests for CNF program, report on the results.)

7. Write a random formula generator for formulas of First Order Logic (as defined in the Week 3 course slides).

8. **Bonus Exercise:** Write a parser for formulas of First Order Logic (as defined in the Week 3 course slides). Use your random formula generator to test the parser.