Object-Oriented Programming

Due: 31 March 2009

1. Find the rule that decides whether a given year is a leap year on wikipedia. Write a program that asks for a number (a year), and that prints `is a leap year` or `is not a leap year`.

2. A *common divisor* of two numbers n_1, n_2 is a number d, s.t. n_1 is divisible by a and n_2 is divisible by a. The greatest common divisor is the biggest such number. Write an C-program that first asks for two numbers and after that finds the greatest common divisor by trying. (Use `n % a == 0` as divisibility test.)

3. You know have the honour of implementing the oldest known algorithm, the *Euclidean Algorithm*. It is based on the following rules:

   (a) If $n_1 < n_2$ then the GCD of n_1 and n_2 equals the GCD of n_2 and n_1.

   (b) If $n_1 \neq 0$, and $n_2 = 0$, then the GCD of n_1 and n_2 equals n_1.

   (c) if $n_1 \neq 0, n_2 \neq 0, n_1 > n_2$, then the GCD of n_1 and n_2 equals the GCD of n_1 and $n_1 - n_2$.

   The Euclidean algorithm computes the greatest common divisor of n_1 and n_2. Check its results against the previous algorithm.

4. Square roots can be computed as follows: Suppose that we want to compute the square root of n. Select two numbers (x_0, x_1), s.t. $x_0^2 \leq n \leq x_1^2$. Compute the average a of x_0, x_1 and check whether $a^2 \leq n$ or $n \leq a^2$. Dependent on the result, replace either x_0 or x_1. Keep on doing this until x_0 and x_1 are sufficiently close together. (For example $x_1 - x_0 < 10^{-10}$.

   (Note that this is just a variation of the number guessing game.)

5. Find some solutions to the following equations, using the method of the previous task:

$$x^3 - 2 = 0,$$

$$x^3 + 2x^2 + 4x - 8 = 0,$$

$$x^2 - 8x + 10 = 0.$$