# CS 70 Discrete Mathematics for CS Spring 2006 Vazirani — HW 4

Please remember to write your section and TA name on your answer sheet.

1. Show that if $a$, $b$, $c$ and $d$ are integers, such that $a|c$ and $b|d$ then $ab|cd$.

2. A common way of producing an infinite sequence of "pseudorandom" numbers $x_0, x_1, x_2, \ldots$ on a computer is to use a *linear congruential generator* as follows: We start with some "seed" value, $x_0$. We then pick integers $a$, $c$ and $m$ and define the rest of the $x_i$'s by the recurrence:

   $$x_{i+1} = (ax_i + c) \bmod m$$

   What is the sequence of numbers produced by this generator when $a = 3$, $c = 1$, $m = 7$ and the seed $x_0 = 4$? How many numbers does this particular generator produce before it starts repeating itself?

3. Use the Extended Euclid Algorithm to find integers $k$ and $l$ such that $\gcd(48, 30) = k \cdot 48 + l \cdot 30$. Show your work.

4. Solve the following equations for $x$ and $y$ or show that no solution exists. Show your work (in particular, what division must you carry out to solve each case).

   (a) $5x + 13 \equiv 11 \pmod{504}$

   (b) $9x + 80 \equiv 2 \pmod{81}$

   (c) The system of simultaneous equations
   $30x + 2y \equiv 0 \pmod{37}$ and $y \equiv 4 + 13x \pmod{37}$

5. **Binary GCD**
   On most computers, the operations of subtraction, testing the parity (odd or even) of a binary integer, and halving can be performed more quickly than computing remainders. This problem investigates the binary gcd algorithm, which avoids the remainder computations used in Euclid's algorithm.

   (a) Prove that if $a$ and $b$ are both even, then $gcd(a,b) = 2gcd(a/2, b/2)$.

   (b) Prove that if $a$ is odd and $b$ is even, then $gcd(a,b) = gcd(a, b/2)$.

   (c) Prove that if $a$ and $b$ are both odd, then $gcd(a,b) = gcd((a-b)/2, b)$ where we assume $a \geq b$.

   (d) Design an efficient binary gcd algorithm that uses $O(\log(max(a,b)))$ subtractions, halving, and parity tests.