Written problems:

1. Let f, g, h be postitive-valued functions, and assume that $h(x) \geq 1$ for all x. Prove that if $f(x) = \mathcal{O}(h(x))$ and $g(x) = \mathcal{O}(1)$, then $f(x) + g(x) = \mathcal{O}(h(x))$ and $f(x)g(x) = \mathcal{O}\left(h(x)\right)$.

2. For any positive integer n, denote by $B(n)$ the number of bits of n. In other words, $B(n)$ is the unique integer such that
$$2^{B(n)-1} \leq n < 2^{B(n)}.$$

   (a) Let $f(n)$ be a function from positive integers to positive real numbers. Prove that $f(n) = \mathcal{O}(n)$ *if and only if* $f(n) = \mathcal{O}\left(2^{B(n)}\right)$.

   (b) Prove that $f(n) = \mathcal{O}(\sqrt{n})$ if and only if $f(n) = \mathcal{O}\left(\sqrt{2}^{B(n)}\right)$.

   (c) Let d be a positive integer. Prove that $f(n) = \mathcal{O}((\log n)^d)$ if and only if $f(n) = \mathcal{O}\left(B(n)^d\right)$.

3. This problem considers a slighlty more general form of babystep-giantstep, in which the babystep list and giantstep list need not have the same length. Let p be a prime number, and let g, h be two units modulo p. Suppose that two positive integers M, N are chosen, and we construct two lists as follows.

   - The babystep list consists of $g^i \mod p$ for $i = 0, 1, \cdots, M-1$.
   - The giantstep list consists of $hg^{-Mj} \mod p$ for $j = 0, 1, \cdots, N-1$.

   (a) Prove that there is a collision between these two lists if and only if there exists a solution x to the discrete logarithm problem $g^x \equiv h \pmod{p}$ with $0 \leq x < MN$.

   (b) Under what circumstances will there be *multiple* collisions between the two lists?

   (c) Suppose that M, N are chosen such that $MN \geq \mathrm{ord}_p(g)$, and further suppose that the lists do not collide. Prove that the discrete logarithm problem $g^x \equiv h \pmod{p}$ has no solution.

4. Solve each system of congruences. Your answer should take the form of a single congruence of the form $x \equiv c \pmod{m}$ describing all solutions to the system.

   (a) $x \equiv 1 \pmod{3}$
   $x \equiv 2 \pmod{5}$

   (b) $x \equiv 6 \pmod{11}$
   $x \equiv 2 \pmod{10}$

   (c) $x \equiv 2 \pmod{3}$
   $x \equiv 1 \pmod{10}$
   $x \equiv 3 \pmod{7}$

   (d) $x \equiv 6 \pmod{8}$
   $x \equiv 3 \pmod{9}$
   $x \equiv 0 \pmod{17}$

Programming problems:

1. (This will be an ingredient in the Pohlig-Hellman algorithm, to be discussed soon) Write a function `ppFactor(N)` which accepts an integer $N \geq 2$, and returns a list of the prime powers (all powers of different primes) factoring N, in any order. For example, if $N = 12$ the function should return either `[4,3]` or `[3,4]`. The integer N may be quite large (up to 1024 bits), but you may assume that all of the prime-power factors are 16 bits or smaller.

   Suggested approach: There are many ways to do this, and certainly many more efficient than what I'm about to describe, but here is one relatively quick-to-implement approach. Write a `for` loop to iterate through all numbers p from 2 to 2^{16}. For each number, check whether it divides N. If so, divide N by p repeatedly until it is no longer divisible by p (and replace N by the new value), then add the appropriate power of p to the list you will eventually return. As long as you shrink N as you go, you will never find that $p \mid N$ unless p is in fact prime, since any smaller factor would have already been found to divide N.

2. Write a function `crtList(ls)` that takes a list `ls` of pairs (a_i, m_i) of integers, with any two of the values m_i relatively prime, and returns a pair (a, m) such that the system of congruences $x \equiv a_i \pmod{m_i}$ is equivalent the single congruence $x \equiv a \pmod{m}$, and $0 \leq a < m$ (i.e. a is reduced modulo m).

   For example, `crtList( [(2,3), (3,5), (0,2)] )` should return $(8, 30)$, since the system of three congruences $x \equiv 2 \pmod{3}, x \equiv 3 \pmod{5}, x \equiv 0 \pmod{2}$ is equivalent to the single congruence $x \equiv 8 \pmod{30}$.

   The integer a should be reduced modulo m, i.e. $0 \leq a < m$. The moduli m_i will be integers up to 256 bits in length, and the list will contain up to 128 entries.