MA/CSSE 473 Day 05 Announcements and Summary

Announcements:

1. HW 2 Due Tonight at 11:55PM.
2. HW3 and HW4 have been updated for this term.

Main ideas from today:

1. If c is a positive constant, find a simple big-Theta expression (as a function of n) for the following sum:
   $f(n) = 1 + c + c^2 + c^3 + \ldots + c^n$

   when $0 < c < 1$

   when $c = 1$

   when $c > 1$

2. Which is harder (computationally): factoring numbers or determining whether numbers are prime?

3. Trace the integer division algorithm from class for `divide(19, 4)`.

4. If x, y and N are k-bit integers, then the time requirement to compute (x + y) (mod N) is Θ ( ).

5. If x, y and N are k-bit integers, then the time requirement to compute (x * y) (mod N) is Θ ( ).

6. When exponentiating n-bit numbers x^y (mod N), where N is also n-bit, how many recursive calls are needed?

7. Each call is Θ( )

8. Entire exponentiation algorithm is Θ( )

9. What problem does Euclid's Algorithm solve?

10. Show the recursive calls for Euclid's Algorithm applied to a=188 and b=144.

11. The following two conditions imply that d = gcd(a,b):

    a.

    b.

12. Prove the validity of the extended Euclid algorithm.

```
def euclidExtended(a, b):
    """ INPUT:  Two integers a and b with a >= b >= 0
        OUTPUT: Integers x, y, d such that d = gcd(a,
b)
                and d = ax + by"""
    print ("    ", a, b) # so we can see the process.
    if b == 0:
        return 1, 0, a
    x, y, d =  euclidExtended(b, a % b)
    return y, x - a//b*y, d
```