Homework 1

1. In each of the following situations, indicate wherether $f = O(g)$, or $f = \Omega(g)$, or both (in which case $f = \Theta(g)$)

	$f(n)$	$g(n)$	O, Ω, or Θ?
(a)	$n - 100$	$n - 200$	$f = \Theta(g)$
(b)	$n^{1/2}$	$n^{2/3}$	$f = O(g)$
(c)	$100n + \log n$	$n + (\log n)^2$	$f = \Theta(g)$
(d)	$n \log n$	$10n \log 10n$	$f = \Theta(g)$
(e)	$\log 2n$	$\log 3n$	$f = \Theta(g)$
(f)	$10 \log n$	$\log(n^2)$	$f = \Theta(g)$
(g)	$n^{1.01}$	$n \log^2 n$	$f = \Omega(g)$
(h)	$n^2/\log n$	$n(\log n)^2$	$f = \Omega(g)$
(i)	$n^{0.1}$	$(\log n)^{10}$	$f = \Omega(g)$
(j)	$(\log n)^{\log n}$	$n/\log n$	$f = \Omega(g)$
(k)	$\sqrt(n)$	$(\log n)^3$	$f = O(g)$
(l)	$n^{1/2}$	$5^{\log_2 n}$	$f = O(g)$
(m)	$n2^n$	$3n$	$f = \Omega(g)$
(n)	2^n	2^{n+1}	$f = \Theta(g)$
(o)	$n!$	2^n	$f = \Omega(g)$
(p)	$(\log n)^{\log n}$	$2^{(\log_2 n)^2}$	$f = O(g)$
(q)	$\sum_{i=1}^{n} i^k$	n^{k+1}	$f = O(g)$

2. Dasgupta Problem 1.13

   Is the difference of $5^{30,000}$ and $5^{30,000}$ a multiple of 31?

   We know that $5^3 = 125$ and $31 * 4 = 124$ so it follows that.

$$5^{30,000} \equiv (5^3)^{10,000} \equiv (125)^{10,000} \equiv 1^{10,000} \equiv 1 (\text{mod } 31)$$

   We also know that $6^6 = 46,656$, $31 \times 1,505 = 46,655$, and $123,456/6 = 20,576$. So it follows that

$$6^{123,456} \equiv (6^6)^{20,576} \equiv (46,655)^{20,576} \equiv 1^{20,576} \equiv 1 (\text{mod } 31)$$

   So it follows that

$$5^{30,000} - 6^{123,456} \equiv 1 - 1 \equiv 0 (\text{mod } 31)$$

   So the difference is divisible by 31.

3. Levitin Problem 2.1.5b

   Prove the alternative formula for the number of bits in the binary representation of a positive integer n:

$$b = \lceil \log_2(n+1) \rceil$$

   We observe that 1) a number $2^{b-1} \mid b \in \mathbb{Z}$ is representable in binary with b bits. We also observe that 2) any number k such that $2^{b-1} \leq k < 2^b$ is also expressible with

b bits. Given a number n, it follows that $n = 2^{b-1} \mid b \in \mathbb{Z}$ or $n \neq 2^{b-1} \mid b \in \mathbb{Z}$. Given the first case, it follows that $n + 1 < 2^b$ unless $n = 0$ or $n = 1$. If $n = 0$ or $n = 1$, then $\lceil \log_2(n + 1) \rceil = 1$, which is the number of bits required to express 0 or 1.