Anooj Pai
Intro to Algo HW1
Collaborators: Shawn Vembenil

1)

a) $f(n) = n - 100 \quad g(n) = n - 200$

Answer: f = Θ(g)

b) $f(n) = n^{1/2} \quad g(n) = n^{2/3}$

Answer: f = O(g)

c) $f(n) = 100n + log(n) \quad g(n) = n + (log(n))^2$

Answer: f = Θ(g)

d) $f(n) = nlog(n) \quad g(n) = 10nlog(10n)$

Answer: f = Θ(g)

e) $f(n) = log(2n) \quad g(n) = log(3n)$

Answer: f = Θ(g)

f) $f(n) = 10log(n) \quad g(n) = log(n^2)$

Answer: f = Θ(g)

g) $f(n) = n^{1.01} \quad g(n) = nlog^2(n)$

Answer: f = Ω(g)

h) $f(n) = \frac{n^2}{log(n)} \quad g(n) = n(log(n))^2$

Answer: f = Ω(g)

i) $f(n) = n^{0.1} \quad g(n) = (log(n))^{10}$

Answer: f = Ω(g)

j) $f(n) = (log(n))^{log(n)} \quad g(n) = \frac{n}{log(n)}$

Answer: f = Ω(g)

2)

T(0)	T(1)	T(2)	T(3)	T(4)	T(5)
1	2	4	8	16	32

We see that there is a pattern with T(n) which is 1,2,4,8,16,32 for the first 6 values of n.

If we look at this pattern we can see that the number of times that a * is printed is 2^n.

We can prove this by using induction:

Base case: T(0) = 1 = 2^0

Induction step:

Assume T(n) = 2^n, now prove that T(n+1) = 2^{n+1}

Let S = the set of T(n) values

S = $2^0 + 2^1 + 2^3 + \dots + 2^n$

Now if we multiply this set by 2 we get 2S which grows to 2^{n+1}

When we subtract these sets: 2S - S we find that they grow to $2^{n+1} - 2^0$

This simplifies to $1 + 2^{n+1} - 1 = 2^{n+1}$

We have proven that T(n+1) is equat to 2^{n+1}

By induction we have proven that T(n) is equal to 2^n

3) We can say that $f(n) \in \theta(g(n))$ if there are constants c, C, and $n_0 > 0$.

If we have these we can conclude $cg(n) \le f(n) \le Cg(n)$.

Because $max(f(n), g(n)) \le f(n) + g(n) \le 2max(f(n), g(n))$ for all n, we can see that $max(f(n), g(n)) = \theta(f(n) + g(n))$

4)

a) Is $2^{2n} = O(2^n)$?
   No it is not equal

b) They are not equal because:
   i) To assume they are equal, c > 0 must exist => $0 \le 2^{2n} \le c * 2^n$ for all n when $n \ge n_0$
   ii) $2^n * 2^n \le c * 2^n$, if we cancel one 2^n from both sides we get $2^n \le c$ which is a contradiction

   By this we have proven that $2^{2n} \ne O(2^n)$