Q1.

(a) \( f(n) = n - 100 \leq 2(n - 100) = g(n) \quad n \geq 100 \)

So, \( f = O(g) \)

On the other hand, \( g(n) = 2n - 200 \leq 2(n - 100) = 2f(n) \quad n \geq 100 \), therefore \( g = O(f) \). Thus, \( f = \Theta(g) \).

(b) \( f(n) = \Theta(n) \) since \( \log n = O(n) \) and the log term can thus be ignored.

Similarly \( g(n) = \Theta(n) \), since \( \log n = O(n^{0.5}) \) (check your class notes), and thus \( \log^2 n = O(n^{0.5n^{0.5}}) \). \( f(n) = \Theta(g(n)) \).

(c) \( g(n) = 10n \log 10n = 10n \log 10 + 10 \log n = \Theta(f(n)) \).

(d) Notice that \( \log(2n^2) = 2 \log n \).

(e) No (e) :)

(f) Using what we have discussed in class, for positive functions, we have \( f_1 \cdot f_2 = O(g_1 \cdot g_2) \) if \( f_1 = O(g_1) \) and \( f_2 = O(g_2) \). Decompose \( f = n^{1.01} \) into \( n \cdot n^{0.005} \cdot n^{0.005} \) and \( g(n) = n \log^2 n \) into \( n \cdot \log n \cdot \log n \). Use the fact that \( \log n = O(n^{0.005}) \).

(g) In class, we also mentioned the fact that for positive functions \( f, g, h \), if \( f = O(g) \), then \( fh = O(gh) \). Here you can decompose \( f(n) \) into \( n \cdot (n/\log n) \) and \( g(n) \) into \( n \cdot (\log n)^3(n/\log n) \). Clearly \( f(n) = O(g(n)) \).

(h) This question is similar to (f). Decompose \( f \) and \( g \) to familiar terms.

(i) This one is similar to (g). Find a common term, \( f(n) = n \cdot 2^n \) and \( g(n) = (3/2)^n \cdot (2^n) \). Since \( n = O((3/2)^n) \), \( f = O(g) \).

(j) Let’s assume that the base is 2. This will not change the complexity order. Notice that \( \log n^2 (\log n)^2 = (2 \log n)^2 \log n = n^{\log n} \). Clearly \( f(n) = O(g) \).

Can you show that this is still true even if the base is not 2?
Q2.

Use the equation: 

\[ 1 + c + c^2 + \ldots + c^n = (c^{n+1} - 1)/(c - 1). \]

Q3.

(a) The base case for \( n = 6 \) is apparently true. Use induction: assume that 

\[ F_k \geq 2^{0.5k} \] 

is true for all \( k \leq n \), let’s prove that \( F_{n+1} \geq 2^{0.5(n+1)} \).

\[ F_{n+1} = F_n + F_{n-1} \geq 2^{0.5n} + 2^{0.5(n-1)} = 2^{0.5n}(1 + 2^{-0.5}) \geq 2^{0.5n} \cdot 2^{0.5} = 2^{0.5(n+1)} \]

We proved it.

(b) Here is our goal

\[ F_{n+1} = F_n + F_{n-1} \leq 2^{cn} + 2^{c(n-1)} = 2^{cn}(1 + 2^{-c}) \leq 2^{cn} \cdot 2^{c} = 2^{c(n+1)} \]

We therefore need to find a \( c \) so that \( 1 + 2^{-c} \leq 2^{c} \). \( c = 0.75 \) is good. Check with your calculator.

(c) This is similar to (b). But we want to find a \( c \) so that \( 1 + 2^{-c} \geq 2^{c} \).

Apparently, the larger the \( c \), the “more unlikely” the inequality will be true. Solve \( x^2 - x - 1 = 0 \) and \( c = \log_2 x \). \( x \approx 0.6942 \).

Q4

Use the method we discuss in class. The sample code is at the course website.