CS200 Quiz 6: Complexity, Recurrences
(10/2/08)

1. [Circle one] What is the smallest Big-O for:
   \[ f(n) = 1 + 4 + 7 + \ldots + (3n + 1) \]
   a. \( \log_2 n \)
   b. \( n \)
   c. \( n^2 \)
   d. \( 2^n \)

2. [Circle one] For what value of \( C \), is
   \[ f(x) = 3 \log x \quad O(x) \text{ given } k=1? \]
   a. 0
   b. 1
   c. 2 either c or d is correct
   d. 3

3. [Circle one] What is \( a_3 \) of the
   recurrence: \( a_n = a_{n-1}^2 \quad , a_1 = 2 \)
   a. 2
   b. 3
   c. 8
   d. 16

4. [Circle all that are true] The terms in the equation \( f(n) = a f(n/b) + g(n) \) mean:
   a. \( a \) is the number of operations done to combine solutions
   b. \( b \) is the number of subproblems formed at each iteration
   c. \( g(n) \) is total number of operations done at each iteration
   d. \( n/b \) is the number of subproblems formed at each iteration

5. [Circle all that are true]
   a. Worst case of an algorithm means the largest number of operations to solve the given problem on input of specified size.
   b. Computational complexity analysis only ever considers time.
   c. Average case analysis is more difficult because you need to know how the algorithm is to be used.
   d. An algorithm has polynomial complexity if it is \( \Theta(n^b), b \geq 1 \)