Informatics 41 • Fall 2006 • Kay/Thornton • UC Irvine Your name ________________________

EIGHTH QUIZ

You have 15 minutes from the start of class to complete this quiz. Read the questions with care; work with deliberate speed. Don't give us more than we ask for. The usual instructions apply. Good luck!

Problem 1 (13 points)

(a) (2 points) Suppose you have an old digital camera that takes only black-and-white pictures (technically, "gray-scale" or "monochrome" pictures, with pixels that can be black, white, or various shades of gray). If the camera uses 4 bits for each pixel, how many different shades of grey can it accommodate (in each pixel, including pure black and pure white)?

(b) (3 points) If the same camera captures an image that's 300 pixels wide and 200 pixels high, how many bytes (not bits) of storage does each image require (assuming no compression)? (Showing your work will help us give you partial credit.)

(c) (2 points) List two reasons why modern computer systems store all their information in bits (using binary circuitry).

(d) (6 points) We saw (at least) three ways to represent numbers in memory: as ASCII characters (8 bits per decimal digit, just like letters and punctuation marks), in BCD (binary coded decimal, with 4 bits per decimal digit), and as binary numbers. For each of these three ways, give *either* one advantage or one disadvantage compared to the other representations. Be sure to say whether each answer is an advantage or a disadvantage.

ASCII:

BCD:

Binary:

Problem 2 (4 points)

Why does adding more memory (RAM) make a computer system faster (i.e., more responsive)? Answer in one clear and concise English sentence, using what you know about computer organization and the storage hierarchy).

Problem 3 (5 points)

For each of the algorithms or operations described below, check the box corresponding most closely to its complexity (i.e., its O-notation) in the average case.

(a) Summing the last three elements of an *n*-element vector of numbers:

❑ Constant—O(1) ❑ Logarithmic—O(log *n*) ❑ Linear—O(*n*) ❑ Quadratic—O(n^2)

(b) In a (balanced) binary search tree of *n* restaurants, ordered by the restaurant's name, producing a list of restaurants alphabetized by name:

❑ Constant—O(1) ❑ Logarithmic—O(log *n*) ❑ Linear—O(*n*) ❑ Quadratic—O(n^2)

(c) In a (balanced) binary search tree of *n* restaurants, ordered by the restaurant's name, finding a restaurant in the tree, given the restaurant's phone number:

❑ Constant—O(1) ❑ Logarithmic—O(log *n*) ❑ Linear—O(*n*) ❑ Quadratic—O(n^2)

(d) In a (balanced) binary search tree of *n* restaurants, ordered by the restaurant's name, finding whether or not a named restaurant is in the tree:

❑ Constant—O(1) ❑ Logarithmic—O(log *n*) ❑ Linear—O(*n*) ❑ Quadratic—O(n^2)

(e) Adding each restaurant on a list of *n* restaurants, one at a time, to a second list of n restaurants, in any order:

❑ Constant—O(1) ❑ Logarithmic—O(log *n*) ❑ Linear—O(*n*) ❑ Quadratic—O(n^2)

Problem 4 (3 points)

One aspect of security in computer systems is authentication—making sure a user is who he or she claims to be. We talked in class about three categories of authentication: What the user *knows,* what the user *has,* and what the user *is.* For each category, give two examples—one in a computer-based system and one in the "real world" (which might be similar to the computer-based one).

"Knows":

"Has":

"Is":