

Name & Section Number: _____

1. For each pair of expressions (A, B) below, indicate whether A is O , o , Ω , ω , or Θ of B . Note that zero, one or more of these relations may hold for a given pair; list all correct ones.

	A	B
(a)	$8n^3 + 14n^2 + 5$	$n^4/5$

(b)	2^n	n^{50}
-----	-------	----------

(c)	$5\sqrt{n}$	$(\log n)^{12}$
-----	-------------	-----------------

(d)	$\sqrt[3]{n}$	$n^{\cos(\pi n/8)}$
-----	---------------	---------------------

(e)	25^n	5^{2n}
-----	--------	----------

(f)	$2^{(n \lg n)}$	10^n
-----	-----------------	--------

(g)	$\log(n!)$	$n \log n$
-----	------------	------------

2. Consider an array of size nine with the numbers in the following order

40, 10, 20, 30, 90, 80, 70, 60, 50.

- (a) Form the heap using the standard (Williams) algorithm. Show the heap as a tree. Show the heap as an array. Exactly how many comparisons did heap creation use?
- (b) Start with the heap created in Part (a). Show the *array* after each element sifts down *after heap creation*. How many comparisons does each sift use? What is the total number of comparisons *after heap creation*?

3. [You will need to wait until after Friday's class to do this problem.] We are going to repeat Problem (2) using Floyd's version for sifting. Consider an array of size nine with the numbers in the following order

40, 10, 20, 30, 90, 80, 70, 60, 50.

- (a) Form the heap using the Floyd's sifting algorithm. Show the heap as an array. Exactly how many comparisons did heap creation use?
- (b) Start with the heap created in Part (a). Show the *array* after each element sifts down using Floyd's sifting algorithm *after heap creation*. How many comparisons does each sift use? What is the total number of comparisons *after heap creation*?

4. We are going to derive a good upper bound on the worst case number of comparisons in (standard) heapsort. In order to simplify the algebra assume that the size of the list that you are sorting is one less than a power of two (i.e., a full tree). Thus, an array of size $n = 2^k - 1$ is associated with a binary tree with k levels numbered $0, 1, \dots, k - 1$.
- (a) How many nodes are on level j ?
 - (b) How many comparisons does the heap formation phase use to sift an element rooted on level j in the worst case.
 - (c) Give a summation for the total number of comparisons for heap formation.
 - (d) Simplify the summation. Show your work.

