Aaron Floreani
CMSI 2120
Homework 5

2. The value of NeedlesslyRecursive.a(4, 2) has 19729 decimal digits:

$$a_n = a_{n-1} + d$$

$$C(n, k) = \frac{n!}{k!(n-k)!}$$

3.

If we expand out the above equation, we get two possible time complexities:
O = (n!)/(k!n!) or O = (n!)/(k!k!)

4.

Selection Sort
[81 9 17 21 20 8 2 5 1 83 23]
Selection sort goes through each element in the array from left to right until it finds the smallest one. Once it finds the smallest value, that item is swapped with the first element of the array. Now that the first element is sorted, it will start at the second element and go right until this second element is swapped. This process repeats until the array is sorted. See below:
[81 9 17 21 20 8 2 5 1 83 23]
[1 9 17 21 20 8 2 5 81 83 23]
[1 2 17 21 20 8 9 5 81 83 23]
[1 2 5 8 20 21 9 17 81 83 23]
[1 2 5 8 9 21 20 17 81 83 23]
[1 2 5 8 9 17 20 21 81 83 23]
…

Gnome Sort
The algorithm looks at the current value and the previous one; if they are in the right order the algorithm shifts to the next value, otherwise, they are swapped and it goes one value back..
[9, 81, 17, 21, 20, 8, 2, 5, 1, 83, 23]
[9, 17, 81, 21, 20, 8, 2, 5, 1, 83, 23]
[9, 17, 21, 81, 20, 8, 2, 5, 1, 83, 23]
[9, 17, 21, 20, 81, 8, 2, 5, 1, 83, 23]
[9, 17, 20, 21, 81, 8, 2, 5, 1, 83, 23]
[9, 17, 20, 21, 8, 81, 2, 5, 1, 83, 23]
[9, 17, 20, 8, 21, 81, 2, 5, 1, 83, 23]
[9, 17, 8, 20, 21, 81, 2, 5, 1, 83, 23]
…
[1, 2, 5, 8, 9, 17, 20, 21, 23, 81, 83]

Insertion Sort

The algorithm is a basic sorting algorithm that sequentially sorts and compares each value from left to right.

[9 81 17 21 20 8 2 5 1 83 23]
[9 17 81 21 20 8 2 5 1 83 23]
[9 17 21 81 20 8 2 5 1 83 23]
[9 17 20 21 81 8 2 5 1 83 23]
[8 9 17 20 21 81 2 5 1 83 23]
[2 8 9 17 20 21 81 5 1 83 23]
…
[1 2 5 8 9 17 20 21 23 81 83]

Quick Sort

QuickSort is also a Divide and Conquer algorithm using recursion. It partitions the array around the chosen pivot.

[9 17 21 20 8 2 5 1 23 83 81]
[1 17 21 20 8 2 5 9 23 83 81]
[1 8 2 5 9 21 20 17 23 83 81]
[1 2 5 8 9 21 20 17 23 83 81]
[1 2 5 8 9 17 20 21 23 83 81]
[1 2 5 8 9 17 20 21 23 83 81]
[1 2 5 8 9 17 20 21 23 81 83]

Merge Sort

The Merge Sort algorithm uses Divide and Conquer to recursively sort. The algorithm continuously splits the array in half until it cannot be further divided, sorting each half.

[81 9 17 21 20 8 2 5 1 83 23]
[81 9 17 21 20 8] [2 5 1 83 23]
[81 9 17] [21 20 8] [2 5 1] [83 23]
[81 9] [17] [21 20] [8] [2 5] [1] [83] [23]
[81] [9] [17] [21] [20] [8] [2] [5] [1] [83] [23]

Now that the array is completely divided, each of the values in the halves will be compared, building back up the array.

[9 81] [17 21] [8 20] [2 5] [1 83] [23]
[9 81] [17 21] [8 20] [2 5] [1 83] [23]
[9 17 21 81] [2 5 8 20] [1 23 83]
[2 5 8 9 17 20 21 81] [1 23 83]
[1 2 5 8 9 17 20 21 23 81 83]

Heap Sort

Heap sort utilizes the Binary Heap data structure to compare nodes. It is similar to the selection sort where we first find the minimum element and place the minimum element at the beginning. Repeat the same process for the remaining elements.
[9 17 21 20 8 2 5 1 23 83 81]
[81 23 17 21 20 8 2 5 1 9 83]
[23 21 17 9 20 8 2 5 1 81 83]
[21 20 17 9 1 8 2 5 23 81 83]
[20 9 17 5 1 8 2 21 23 81 83]
[17 9 8 5 1 2 20 21 23 81 83]
[9 5 8 2 1 17 20 21 23 81 83]
…
[1 2 5 8 9 17 20 21 23 81 83]

Radix Sort
Radix sort is a sorting algorithm that sorts the elements by first grouping the individual digits of the same place value, like we did in class. Then, sort the elements according to their increasing or decreasing order.
[20 81 17 21 20 8 2 5 1 83 23]
[20 81 17 21 20 8 2 5 1 83 23]
[20 81 21 21 20 8 2 5 1 83 23]
[20 81 21 1 20 8 2 5 1 83 23]
[20 81 21 1 2 8 2 5 1 83 23]
[20 81 21 1 2 83 2 5 1 83 23]
[20 81 21 1 2 83 23 5 1 83 23]
[20 81 21 1 2 83 23 5 1 83 23]
[20 81 21 1 2 83 23 5 17 83 23]
[20 81 21 1 2 83 23 5 17 8 23]
[20 81 21 1 2 83 23 5 17 8 9]
…
[1 2 5 8 9 17 20 21 23 81 83]

5. Sorting algorithms description:
Exchange: compares each element of an array and swaps the values to a more sorted position.
Insertion: values are moved from an unsorted part of the list to a sorted part of the list in their correct order.
Selection: finds the minimum value from the unsorted part and puts it at the beginning of the sorted part.
Merge: uses recursion to divide and sort the smallest sections and then rebuilds those sorted sections into the full list again.
Distribution: first the values are distributed into ranks such as a binary tree and then they're sorted.
Hybrid: hybrid sorts combine two or more sorting algorithms to sort.

Concurrent: parallel sorting algorithm that compares values in a predefined sequence and the sequence of comparison doesn't depend on data
Impractical: these silly algorithms are made for fun or as a challenge to find slow ways to sort

7. **What is the size of this tree?**
16
How many edges does this tree have?
15
What are the leaves?
H, N, F, C, J, O, P, L, M
What are the children of D?
G
What is the depth of G?
2
What is the degree of G?
4
What are the ancestors of G?
D, A
What are the descendants of G?
J, K, L, M, O, P
What are the nodes on level 3?
H, I, J, K, L, M
What is the height of the tree?
4
What is the width of the tree?
6
What is the height of the node D?
3
What is the simplest path from P to E?
P - K - G - D - A - B - E
What is the degree of the tree?
4
Enumerate the nodes in breadth-first order.
A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P
Enumerate the nodes in depth-first order.
A, B, E, H, I, N, F, C, D, G, J, K, O, P, L, M

8.