Algorithms Analysis Problems 1

Alex Eschenauer

1. What is the order of the following growth functions?
   1. 10n^2 + 100n + 1000
      1. ANSWER: n^2
   2. 10n^3 - 7
      1. ANSWER: n^3
   3. 2^n + 1000n^3
      1. ANSWER: 2^n
   4. n^2 log n
      1. ANSWER: n log n
2. Arrange the growth functions of the previous exercise in ascending order of efficiency for n = 10 and again for n = 1,000,000.
   1. N=10 in ascending order:
      1. d, a, b, c
   2. N=1000000 in ascending order:
      1. d, a, b, c
3. Write the code necessary to find the largest element in an unsorted array of integers. What is the time complexity of this algorithm?

-------------------------------- CODE: -----------------------------------

int[] arrayOfIntegers = {randomint1, randomint2, randomint3}; //pretend that this is full of ints

int largest;

for (int i=1; i<arrayOfIntegers.length; i++) {

if (arrayOfIntegers[i-1] >= arrayOfIntegers[i]) { largest = arrayOfIntegers[i-1]; }

}

System.out.println("Largest : " + largest);

-------------------------- ANSWER TO SECOND PART: -----------------------------------

Time complexity is n, because since complexity is based off of the power of the function, and the power is simply zero because it is only integers, so that means that time complexity is n.

---------------------------------------------------------------------