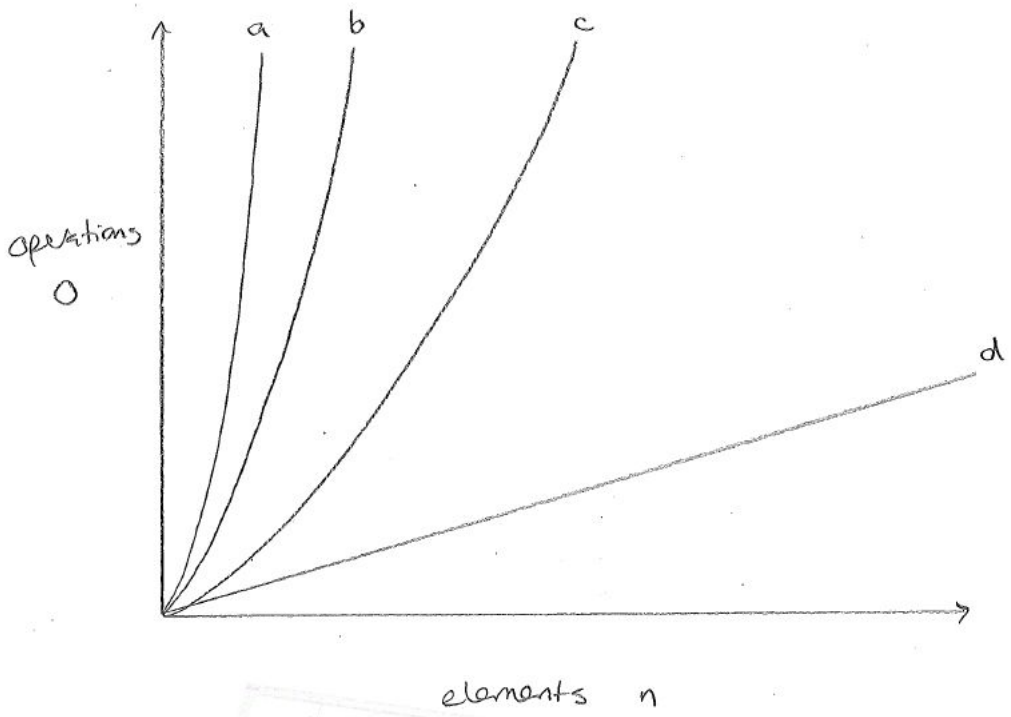
MATH ~~155~~ 156 – Chapter 4 Practice Questions

1. (4 points) Match the Big O notation with its corresponding curve on the graph.



(a) $O(n^2)$ c

(b) $O(n!)$ a

(c) $O(n)$ d

(d) $O(2^n)$ b

2. (5 points) Suppose you are trying to find an entry in an ordered list. You try two different methods:

   - Method 1: You start at the beginning of the list and go down until you find the entry you want. This has $O(n)$.
   - Method 2: You go to the halfway point and deterine whether the entry of interest is above or below the that middle entry. Then divide that part of the list in half and check the halfway point. Repeat until you've found the entry of interest. This is called a binary search and has $O(\log n)$.

   Answer the following questions about the above scenario.

   (a) If the list has only 10 items and you are not using a computer for this task, then the most efficient method is probably __#1__

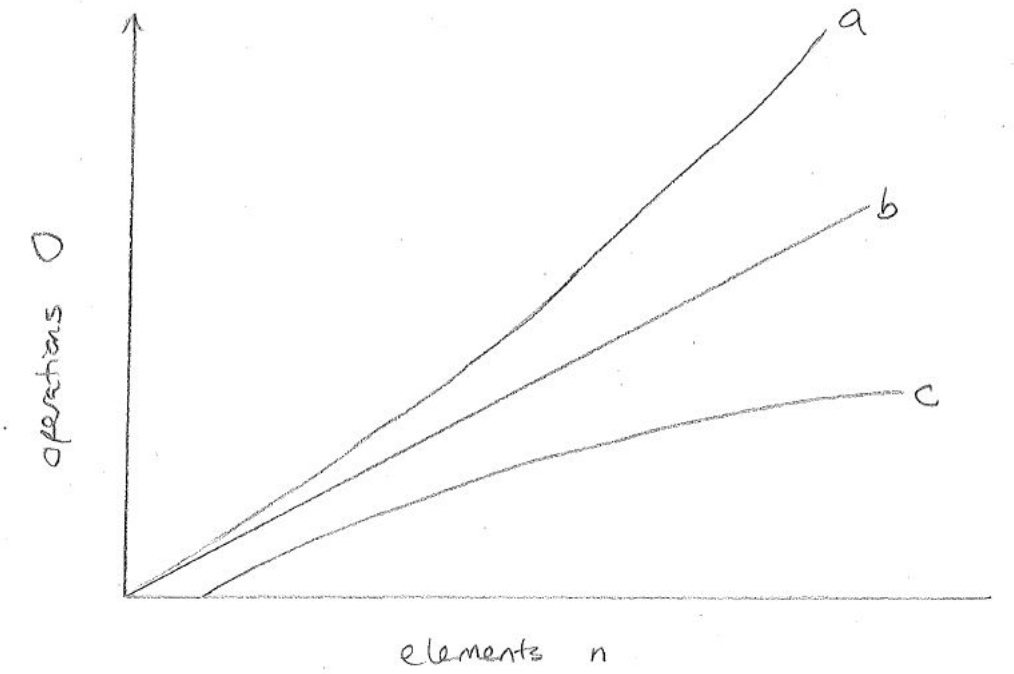
   (b) If the list is very long, then the most efficient method is definitely __#2__.

   (c) For method 1, the best case scenario is that the entry you want is located in the following place: (top) / middle / bottom of the list

   (d) For method 1, the worst case scenario is that the entry you want is located in the following place: top / middle / (bottom) of the list

   (e) For method 2, the best case scenario is that the entry you want is located in the following place: top / (middle) / bottom of the list

3. (3 points) Match the Big O notation with its corresponding curve on the graph.

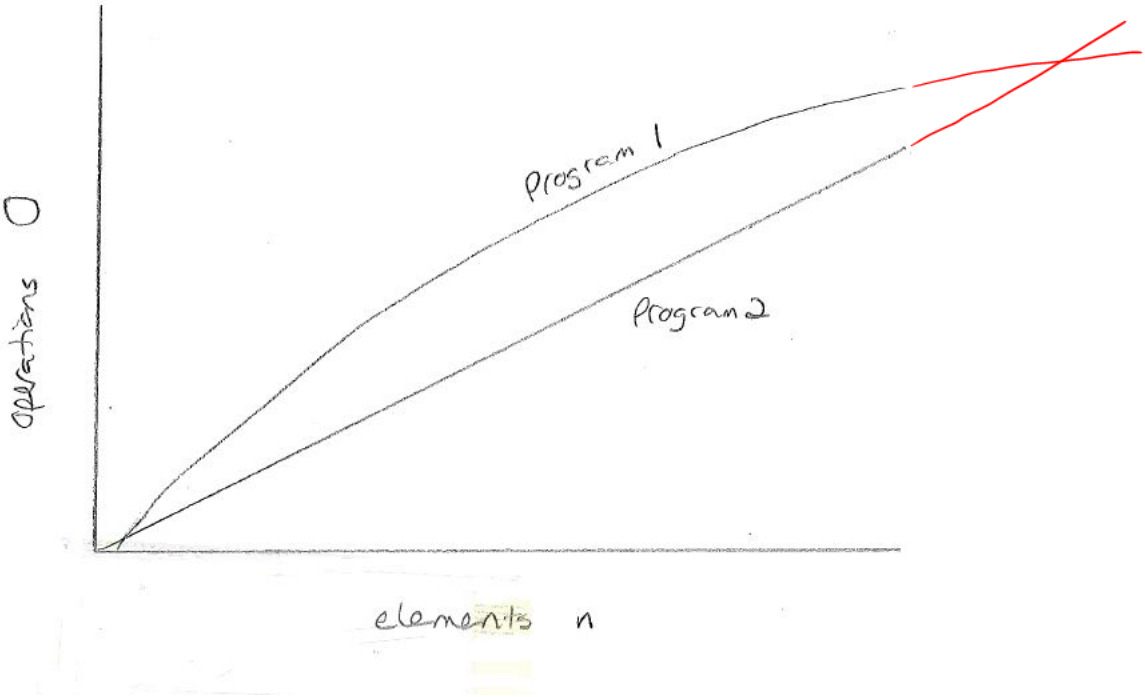


(a) $O(\log n)$ c

(b) $O(n \log n)$ a

(c) $O(n)$ b

4. (4 points) This graph shows the number of operations O required to complete a task of size n elements for Programs 1 and 2, where Program 1 is the curved line and Program 2 is the straight line.



Indicate whether the following statements are true or false.

(a) Program 1 could be $O(2^n)$. True / (False)

(b) Program 1 could be $O(\log n)$. (True) / False

(c) Program 2 could be $O(n \log n)$. n log n is not a straight line True / (False)

(d) For large n, Program 2 will finish faster because the line for Program 2 is below the line for Program 1 at the right-hand side of the graph. True / (False)

If you continue the lines off the right-hand-side, you can see that the lines cross almost immediately and Program 1 has fewer operations required beyond that point.