CSD3130 Tutorial 2

Question 1

A sorting method with "Big-Oh" complexity $O(n\log_{10}n)$ spends exactly 1 millisecond to sort 1,000 data items. Assuming that time T(n) of sorting n items is directly proportional to nlog10n, that is, $T(n)=(cn\log_{10}n)$, where c is a constant. Derive a formula for T(n), given the time T(N) for sorting N items, and estimate how long this method will sort 1,000,000 items.

T(n)= cnlog10n
n = 1000
c = 1/1000log(1000)
c = 1/3000

T(10^6) = 2000m/s = 2sec

Question 2

Assume that each of the expressions below gives the processing time T(n) spent by an algorithm for solving a problem of size n. Select the dominant term(s) having the steepest increase in n and specify the lowest Big-Oh complexity of each algorithm.

Expression	Dominant term(s)	Big-Oh complexity
$5+0.0001n^3+0.025n$	0.0001n^3	O(n^3)
$500n+100n^{1.5} + 50n\log_{10}n$	100n^1.5	O(n^1.5)
$n^2\log2n+n(\log2n)^2$	n^2log2n	O(n^2log2n)

Question 3:

The number of operations executed by algorithms A and B is $8n \lg n$ and $2n^2$, respectively. Determine n^0 such that A is better than B for $n \geq n^0$.

n = 17

When n = 16, both algorithm will meet at the same point. After that, A will be better than B