CS430 Assignment 1
Due: May 26th, 2024

(Ethics: Any behavior on any homework or exam that could be considered copying or cheating will result in an immediate zero on the assignment for all parties involved and will be reported to academichonesty@iit.edu. See the IIT Code of Academic Honesty, https://web.iit.edu/student-affairs/handbook/fine-print/code-academic-honesty)

Use pseudo-code in questions that need you to present your algorithms, but your pseudo-code can be "English-like" if the operation is obvious. Make sure that each line in your pseudo-code is numbered, and the indentation is correct.

1. Arrange the following functions of n (you may assume the domain of variable n is $\mathbb{N}$) so that each function is big-Oh of the next function. In other words, order them by their speed of growth in non-decreasing order. Group together those functions that are of the same order.

$6n \times \log_2 n$	2^$\log_2 n$	e^n	2^{100}	$\log_2 \log_2 n$
$\ln n^2$	2^2^n	n^3	$n^{1/100}$	$4 \text{ x } n^{3/2}$
$\log n^{100}$	4^n	$2^{2\ \wedge \log 2\ n}$	$\sqrt{n^3}$	$n^{0.5}+1$
$n \log_4 n$	$\log_2{}^2 (n)$	$100^{100\wedge 100}\, n$	$4^{\log_2 n}$	$n^2 \log n$
1	$n \cdot 2^n$	$\log_2 (n/2)$	$n \,.\log_2{}^2 (\text{n})$	$\sqrt{\log n}$

2. Consider an alternative version of the merge sort: instead of dividing an array into two sub-arrays evenly, we now divide the array into three sub-arrays evenly.
   a) How to merge three sorted arrays? What is the worst-case time complexity of your merge-three-arrays algorithm?
   b) Present pseudo code for the alternative version of the merge sort and analyze its time complexity.

3. Let $A[1 \ldots n]$ be an array of n distinct numbers. If $i < j$ and $A[i] > A[j]$, then we say (i, j) is an $inversion$ of A. Answer the following questions.
   a) What are the inversions in array $\{1,4,2,9,6,3\}$?
   b) Consider the same array as in question a). Denote the number of inversions within the first half-array (aka $\{1,4,2\}$) as a, then denote the number of inversions within the second half-array (aka $\{9,6,3\}$) as b, and denote the number of inversions crossing two halves as c. What are the values of a, b, c? What's your observation about values a, b, c and the answer of part a)?
   c) Consider the array in part a) and the values a, b, c in part b). If I only change the orderings inside both halves, which value among a, b, c won't change?
   d) Present an algorithm to calculate the number of inversions of $A[1 \ldots n]$ with n distinct numbers by modifying merge sort. Hint: You can use the observations in parts a), b) and c).
   e) Analyze the time complexity your algorithm presented part d).

4. Let $A[1 \ldots n]$ and $B[1 \ldots n]$ be two sorted arrays.
   a) Present an algorithm to find the median of all $2n$ elements in arrays A and B. To get full marks, your algorithm should have a worst-case time complexity $\Theta(\lg n)$. Hint: This algorithm can be inspired by the algorithm in Lecture 4.
   b) Analyze the time complexity your algorithm presented part a).