CS 430 – S 2024
INTRODUCTION TO ALGORITHMS
HOMEWORK #1
DUE JAN 23

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

- Assignment Instruction
  - Team work is NOT allowed.
  - Submit your answers in PDF version to the Blackboard.
  - No late submission accepted.
  - All solutions should be explained.

1. Suppose we are comparing implementations of insertion sort and merge sort on the same machine. For inputs of size n, insertion sort runs in $6n^2$ steps, while merge sort runs in 42nlgn steps. Do you think merge sort alway prevails on insertion sort? If not, for which values of n does insertion sort beat merge sort? (2pts)

2. What is the time complexity of each code below? Show your work. (3pts)

(1)

```
count=1
while count<n
    count=count*3
```

T(n)=

(2)

```
i=0
for i<n
    j=0
    for j<n
        j=j*2
    i++
```

T(n)=

(3)

```
i=0
for i<n
    j=i
    for j<=i and j<n
        j++
    i++
```

T(n)=

3. Examine each function in the left column. Determine the functions listed in the first row are their lower bound, upper bound or asymptotic bound by filling in with the corresponding symbols. If multiple relations apply, choose the most rigorous one. The second row shows an example. (3pts)

	n	n^2	2^n
$example$: n^2	**Ω**	**θ**	**O**
n^n			
$nlog\, n^4$			
$(n - 2)!$			
2^{logn}			

4. To deal with the case that one subarray is shorter than the other one, when we design MERGE (A, p, q, r) to merge L and R into a sorted array, at the end of each subarray, we add an extra element ∞ to serve as a sentinel. Rewrite MERGER without using sentinels. Instead, the rewritten algorithm should stop comparison when all elements from either L or R are all done and copy the reminder of the other array into A. (2pts)