Theory Assignment 1: CS2233

August 7, 2025

Kindly adhere to the following instructions.

- Please neatly write your solutions on the answer sheet. Please clearly mention your name and roll number in the top right corner of the paper. The solutions should be self-explanatory with required mathematical arguments/proofs.
- Please submit the scanned copy of the solution to the google classroom portal and hardcopy to the classroom.
- Any form of plagiarism (web/chatGPT/with peers) will be severely penalised and will result in F grade.
- The submission (strict) timeline is 21nd August, Thursday, 12 NOON (after the class).

---

1. For each of these questions, briefly explain your answer. 3 Marks
   - If I prove that an algorithm takes $O(n^2)$ worst-case time, is it possible that it takes $O(n)$ on some inputs?
   - If I prove that an algorithm takes $O(n^2)$ worst-case time, is it possible that it takes $O(n)$ on all inputs?
   - If I prove that an algorithm takes $\Theta(n^2)$ worst-case time, is it possible that it takes $O(n)$ on some inputs?
2. Let $f(n)$ and $g(n)$ are non-negative functions over non-negative input. Prove or disprove the following and justify your claims via formal mathematical arguments.
   8 Marks
   - $f(n) + o(f(n)) = \Theta(f(n))$.
   - $f(n) = O(g(n))$ implies $g(n) = \Omega(f(n))$.
   - $f(n) + g(n) = \Theta(\min\{f(n), g(n)\})$
   - $n! = o(n^n)$ and $n! = \omega(2^n)$

3. Is the function $\lceil \ln n \rceil !$ polynomially bounded? Is the function $\lceil \ln \ln n \rceil !$ polynomially bounded? 3 Marks

4. Solve the following recursions. Mention any assumptions you might be making. 5 Marks

   - $T(n^2) = 7T(n^2/4) + cn^2$ and $T(1) = 1$
   - $T(n) = n * T(\sqrt{n})$ and $T(2) = 4$ (Assume n to be of the form 2^{2^i})
   - $T(n) = T(n/2) + 2T(n/4) + 3n/2$ (whenever $n > 3$) and $T(1) = 0, T(2) = 2$
   - $T(n) = 4T(n/2) + n^3$ and $T(1) = 1$
   - $T(n) = T(n/2) + c \log n$

5. Arrange the following function by order of growth. That is, find an ordering among the following functions $f_1, f_2, f_3, \ldots, f_6$ such that $f_1 = \Omega(f_2), f_2 = \Omega(f_3), \ldots f_5 = \Omega(f_6)$. Partition the list into equivalence class such that functions $f(n)$ and $g(n)$ are in the same class (that is, they are of the same order), if and only if $f(n) = \Theta(g(n))$. Justify your claims.

   3 Marks

   - $\frac{n^{1.2}}{\log n}$
   - n^2
   - $n \log n$
   - 1.1^n
   - 0.9^n
   - $\log^3 n$

6. What value is returned by the following function? Express your answer as a function of n. Give the worst-case running time using big-Oh notation. 3 Marks

```
function XYZ(n)
r := 0
    for i := 1 to n do
        for j := 1 to i do
            for k := j to i + j do
                for l := 1 to i + j − k do
                    r := r + 1;
return(r)
```