CS3230 Semester 1 2025/2026
Written Assignment 1
(22 August – 05 September, 2025)

Note: All the logarithms are of base 2, that is, $\log x = \log_2 x$.

Question 1: Asymptotic Analysis ($10 \times 2.2 = 22$ marks)

For each of the following questions, you are given two functions $f(n)$ and $g(n)$, and you need to answer one of the following:

- $f(n) \in o(g(n))$.
- $f(n) \in \Theta(g(n))$.
- $f(n) \in \omega(g(n))$.
- None of the above is True.

For example, if $f(n) = n^2$ and $g(n) = n^3$, the correct answer is $f(n) \in o(g(n))$.

Optional: You may give a brief explanation or complete proof for your answers. If you include one, the TA will offer feedback. There is no penalty for incorrect explanations or proofs.

1. $f(n) = 3n + 1$, $g(n) = 4n$.
2. $f(n) = 3^n$, $g(n) = 2^n \cdot n^{10000}$.
3. $f(n) = 2^n$, $g(n) = 2^{n+1}$.
4. $f(n) = 2^{(2^n)}$, $g(n) = 2^{\left(2^{n+1}\right)}$.
5. $f(n) = \sum_{i=1}^{n} \frac{1}{2^i}$, $g(n) = \sum_{i=1}^{n} \frac{i^{10000}}{2^i}$.
6. $f(n) = \log\log(n^n)$, $g(n) = \log\log\left(2^{\sqrt{n}}\right)$.
7. $f(n) = 2^{\sqrt{\log n}}$, $g(n) = \sqrt{n}$.
8. $f(n) = 2^{\sqrt{\log n}}$, $g(n) = 2^{\left(2^{\sqrt{\log\log n}}\right)}$.
9. $f(n) = n^{\frac{1}{\log n}}$, $g(n) = n^{\frac{1}{(\log n)^2}}$.
10. $f(n) = n^{\frac{1}{\log n}}$, $g(n) = n^{\frac{1}{\sqrt{\log n}}}$.

Question 2: Recurrences ($5 \times 4.2 = 21$ marks)

Solve the following recurrence relations by deriving an asymptotically tight upper bound in $O(\cdot)$ notation. You only need to prove the upper bound. While no lower bound proof is required, the bound should be tight.

If you apply the Master Theorem, clearly indicate which case applies and show the calculations that justify its use. If the Master Theorem does not apply, give a detailed derivation using another method, such as telescoping, recursion trees, or the substitution method.

As usual, you may assume there exists a constant n_0 of your choice such that the recurrence relation applies only for $n \geq n_0$. For the base case, you may assume there is a constant C such that $T(n) \leq C$ whenever $n < n_0$.

1. $T(n) = 16 \cdot T(n/4) + 32 \cdot n \log^{128} n$.
2. $T(n) = 16 \cdot T(n/4) + 64 \cdot n^2 \log^8 n + 32 \cdot n \log^{128} n$.
3. $T(n) = 16 \cdot T(n/4) + n^4$.
4. $T(n) = T(\sqrt{n}) + \log n$.
5. $T(n) = T(\sqrt{n}) + \log \log n$.