University of Colombo School of Computing
SCS 1308 - Foundations of Algorithms

Tutorial - 01

1. **Use the substitution method** to show that for the recurrence equation:

   T(1) = 8
   T(n) = T(n−1)+4n
   The solution is T(n) = $\Theta(n^2)$.

2. **Use the substitution method** to show that for the recurrence equation:

   T(1) = 1
   T(n) = T(n/2)+n
   The solution is T(n) = $\Theta(n)$.

3. **Use the recursion tree method** to find an asymptotic upper bound for the recurrence equation:

   T(n) = T(n/2) + n^2
   Use the substitution method to prove your answer.

4. Solve the following recurrence relation using the **Substitution method**:

   T(N) = 2T(N−1)+1, with T(0) = 0.

5. Solve the given recurrence equation by **Master's method**. Justify your solution clearly.

   1. Assume that $n = 2^m$, where $m \geq 0$:

   $$T(n) = \begin{cases} 1, & n < 2 \\ 3T(n/2) + n^2, & n \geq 2 \end{cases}$$

   2. Assume that $n = 3^m$, where $m \geq 0$:

   $$T(n) = \begin{cases} 1, & n < 3 \\ 4T(n/3) + n, & n \geq 3 \end{cases}$$

   3. Assume that $n = 4^m$, where $m \geq 0$:

   $$T(n) = \begin{cases} 1, & n < 4 \\ 16T(n/4) + n^2 \log n, & n \geq 4 \end{cases}$$

6. Solve the following recurrence relation using **Master's theorem.**

$$T(n) = \begin{cases} 1 \quad , & \text{if } n \leq 2 \\ 2T\left(\frac{n}{2}\right) + n\log n, & \text{if } n > 2 \end{cases}$$

7. Solve the following recurrence relation using **Iteration's Method**

    a. $T(n) = T(n-1) + n^2$ $\qquad T_1 = 2$

    b. $T(n) = 2T(n/2) + 3$ $\qquad T_1 = 1$