Theepakorn Phayonrat 67011352

Task 3: Complexit Analysis on Recursion (Do not use master method)

3.1 Show time complexity of $T(n) = 2T(\frac{n}{2}) + n$

Solution

$$
\begin{aligned}
T(n) &= 2(T(\frac{n}{2})) + n \\
&= 2(2T(\frac{n}{2^2}) + \frac{n}{2}) + n \\
&= 2(2T(\frac{n}{2^2})) + n + n \\
&= 2^2T(\frac{n}{2^2}) + 2n \\
&= 2^3T(\frac{n}{2^3}) + 3n \\
&= 2^4T(\frac{n}{2^4}) + 4n \\
&. \\
&(Continuektimes) \\
&. \\
T(n) &= 2^kT(\frac{n}{k}) + kn
\end{aligned}
$$

$$
\begin{aligned}
T(1) &= 1 \\
T(n) &= T(1) \\
\frac{n}{2^k} &= 1 \\
n &= 2^k \\
k &= log_2 n
\end{aligned}
$$

$$
\begin{aligned}
T(n) &= 2^kT(\frac{n}{k}) + kn \\
&= n(T(1)) + nlog_2 n \\
&= n + nlog_2 n
\end{aligned}
$$

Answer

$\therefore T(n) = O(nlog(n))$

Theepakorn Phayonrat 67011352

3.2 Show time complexity of $T(n) = T(n-1) + 1$

Solution

$$
\begin{aligned}
T(n) &= (T(n-2)+1)+1 \\
&= T(n-2)+2 \\
&= T(n-3)+3 \\
&= T(n-4)+4 \\
&\cdot \\
&(Continuektimes) \\
&\cdot \\
T(n) &= T(n-k)+k
\end{aligned}
$$

$$
\begin{aligned}
T(0) &= 1 \\
\text{Assume } n-k &= 0 \\
\therefore n &= k \\
\therefore T(n) &= 1+n
\end{aligned}
$$

Answer

$\therefore T(n) = O(n)$