1-a)

depth

$1 \cdot 1$

$1 + 2 \cdot 2^1$

$1 + 2 \cdot 2^1 + 3 \cdot 2^2$

$1 + 2 \cdot 2 + 3 \cdot 2^2 + 4 \cdot 2^3$

$\vdots$

$$= \sum_{i=1}^{h} 2^{i-1} \cdot i = 2^h \cdot (h-1) + 1$$

$min = 2^h(h-1) + 1 - h$ (In the complete binary tree, there may be only one child)

$max = 2^h(h-1) + 1 - (2^{h-1} - 1) \cdot h$ (In the complete binary tree, there may be at most one less then the perfect binary tree)

$$\text{Avarge} = \frac{min + max}{2}$$

$$= \frac{\left(2^h(h-1) + 1 - h\right) + \left(2^h(h-1) + 1 - (2^{h-1} - 1) \cdot h\right)}{2}$$

$$= \underline{2^h(h-4) - 15}$$

1-b/ $T(n) = T(n/2) + 1$

$T(n/2) = T(n/2^2) + 1$

- $T(n) = T(n/2^2) + 1 + 1$

$T(n/2^2) = T(n/2^3) + 1$

- $T(n) = T(n/2^3) + 1 + 1 + 1$

$\vdots \qquad \vdots \qquad \vdots$

- $T(n) = T(\underbrace{n/2^k}_{1}) + k$

$k = \log n$

$T(n) = 1 + \log n \Rightarrow \log n$

1-C) Yes There some restrictions:

<u>Theorem 1:</u> let T be a binary tree. For every $k \geq 0$, there are no more than 2^k nodes in level k

<u>Theorem 2:</u> let T be a binary tree wit λ levels, then T has no more than $2^\lambda - 1$ nodes.

N : nodes
L : num of Leaves
I : num of internal nodes

> If we know any one of them, we can determine the other two.

let T be a nonempty, full binary tree;

→ If T has I internal nodes, the number of leaves is $\boxed{L = I + 1}$

$\boxed{N = 2I + 1}$

→ If T has a total of N nodes:

↳ $I = \frac{N-1}{2}$

↳ $L = \frac{N+1}{2}$

→ If T has L leaves

↳ $N = 2L - 1$

↳ $I = L - 1$