Handout 6 **January 29, 2020**

CSci 242: Algorithms and Data Structures Instructor: Dr. M. E. Kim

Properties of Proper Binary Tree (slide chap2-#45)

In the proper binary tree T, each symbol denotes a following number:

- n: the number of nodes in T,
- e: the number of **external** nodes in T,
- i: the number of **internal** nodes in T.
- h: the **height** of T.

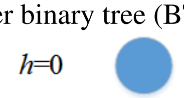
1). $e = i + 1$:

Proof: Prove it by induction on the height h of T.

Base Case:

i) $h = 0$:

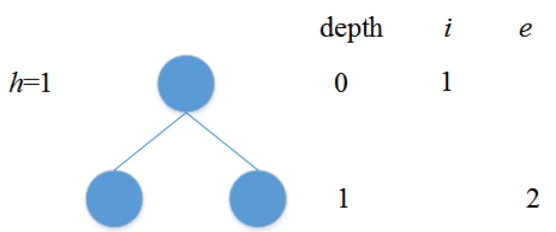
A proper binary tree (BT) with a single node which is a root of T.



Since T has the root only which an external node, $e = i + 1$ holds in T of height $h = 0$.

ii) $h = 1$:

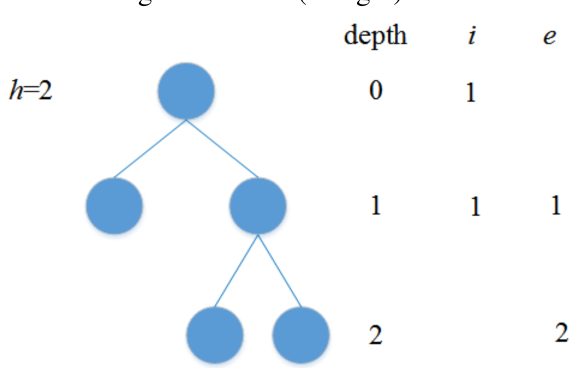
A proper BT of height 1 is only when a root has both left child and right children.



Since T has 2 external nodes and 1 internal node (i.e. root), $e = i + 1$ holds in T of height $h = 1$.

iii) $h = 2$:

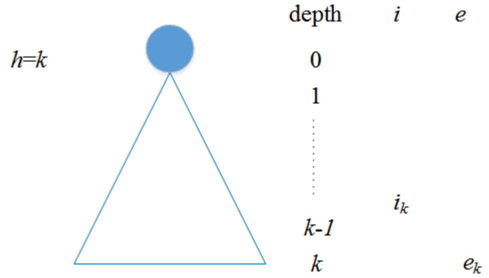
A proper BT of height 2 is a left (or right) child of a root has both left child and right children.



Since T has 3 external nodes and 2 internal node (i.e. root), $e = i + 1$ holds in T of height $h = 2$.

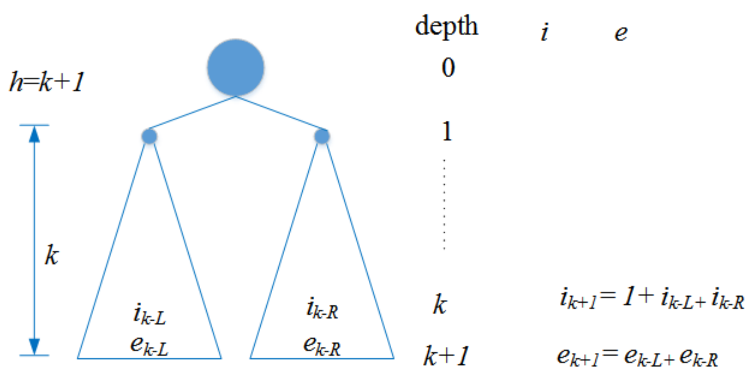
Inductive Hypothesis:

Assume that $e = i + 1$ *holds* in the proper binary tree of height k : i.e. $e_k = i_k + 1$.



Inductive Step:

Let's prove that $e = i + 1$ holds in the proper binary tree of height $k + 1$ using the Inductive Hypothesis.



A proper binary tree T of height $h = k + 1$ is a tree in which both left-subtree and/or right-subtree of a root are proper binary tree of height k. Let

- e_{k-L} : the number of external nodes in T's left-subtree of height k,
- e_{k-R}: the number of external nodes in T's right-subtree of height k,
- i_{k-L} : the number of internal nodes in T's left-subtree of height k,
- i_{k-R}: the number of internal nodes in T's right-subtree of height k,
- e_{k+1}: the number of external nodes in T, and
- i_{k+1}: the number of internal nodes in T.

Since the number of internal nodes of a proper BT of height $k + 1$ is equal to a sum of the internal nodes of its left and right subtrees plus one which is a root,

$$i_{k+1} = i_{k-L} + i_{k-R} + 1. \quad -* \text{ (eq. 1)}$$

Similarly, the number of external nodes of a proper BT of height $k + 1$ is equal to a sum of the external nodes of its left and right subtrees,

$$e_{k+1} = e_{k-L} + e_{k-R}. \quad -** \text{ (eq. 2)}$$

Thus, $e_{k+1} = e_{k-L} + e_{k-R}$ by ** (eq. 2)

$= (i_{k-L} + 1) + (i_{k-R} + 1)$ by **Inductive Hypothesis**

$= (i_{k-L} + i_{k-R} + 1) + 1$

$= i_{k+1} + 1.$ by * (eq. 1)

Hence, $e = i + 1$ also holds in a proper BT of height $k + 1$.

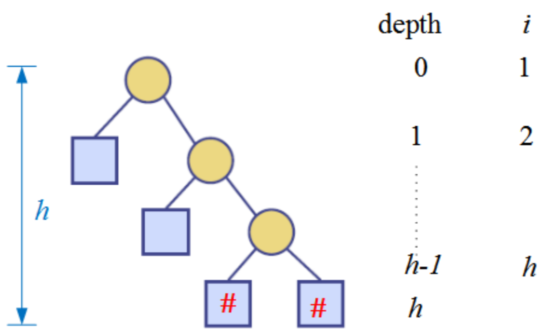
Therefore, in the proper binary tree of any height, $e = i + 1$. **Q.E. D.**

2). $\boldsymbol{n = 2e - 1}$:

Proof:
Since $n = e + i$ in T, $n = e + (e - 1) = 2e - 1$ where $i = e - 1$ in **1**.
Thus, $n = 2e - i$. Q.E.D.

3). $\boldsymbol{h \leq i}$:

Proof:
The height of a proper BT is the *highest* is when only a left (or right) child of a root of any subtree keeps expanding its two children while any right (or left) child doesn't expand a child at all, and vice versa. For example, a right child of a root of each subtree keeps expanding its two children but any left child does not in Figure 1 below.



In such a case, the number of internal nodes is equal to the height of a proper BT because a root is included as an internal nodes while the external nodes at the height of tree (#) are excluded when i is counted.
Thus, the height of a proper BT is at most the number of internal nodes.
Thus, $h \leq i$. Q.E.D.

Proof:
4). $\boldsymbol{h \leq \frac{(n-1)}{2}}$:

Proof:
In **2**, $n = 2e - 1$. Since $n = 2e - 1 = 2(i + 1) - 1 = 2i + 1$, $i = \frac{n-1}{2}$.
Thus, $h \leq \frac{n-1}{2}$ in **3**. Q.E.D.

5). $\boldsymbol{e \leq 2^h}$:

Proof:
The maximum number of external node of a proper BT happens when a proper BT is *complete*, i.e. all internal nodes have exactly two children and all external nodes have the same depth. (Refer to slide #12 & #14 of Appendix _B_Tree.)
In such a case, the number of nodes at each depth d is 2^d, i.e. the number of external nodes at the depth h is 2^h.
Thus, the number of external nodes of complete BT of height h is at most 2^h,
Hence, $e \leq 2^h$. Q.E.D.

6). $\boldsymbol{h \geq \log_2 e}$:

Proof:
In **5**, $e \leq 2^h$. Apply the logarithm of base 2 to both sides.

$$\begin{aligned}\log_2 e \leq \log_2 2^h &\Leftrightarrow \log_2 e \leq h \cdot \log_2 2 \\ &\Leftrightarrow \log_2 e \leq h \cdot 1 \\ &\Leftrightarrow \log_2 e \leq h. \quad \text{Q.E.D.}\end{aligned}$$

7). $\boldsymbol{h \geq \log_2 (n+1) - 1}$:

Proof:
In **2**, $n = 2e - 1$, i.e. $e = \frac{n+1}{2}$.

Let us apply it to the above **6**.

$$\log_2 e \leq h \Leftrightarrow \log_2 \frac{n+1}{2} \leq h$$

$$\begin{aligned}&\Leftrightarrow \log_2 (n+1) - \log_2 2 \leq h \\ &\Leftrightarrow \log_2 (n+1) - 1 \leq h. \quad \text{Q.E.D.}\end{aligned}$$