# B-Tree Height Complexity Proof

Shyamal Suhana Chandra



## 1 Theorem: B-Tree Height

**Statement:** For a B-Tree of order $t$ (min degree) with $n$ keys, the height $h$ satisfies:

$$h \leq \log_t \frac{n+1}{2}$$

## 2 Proof

A B-Tree of height $h$ has at least:

- Root: 1 node with at least 1 key
- Level 1: at least 2 nodes with at least $(t-1)$ keys each
- Level 2: at least $2t$ nodes with at least $(t-1)$ keys each
- Level $h$: at least $2t^{h-1}$ nodes with at least $(t-1)$ keys each

Total minimum keys:

$$n \geq 1 + 2(t-1) + 2t(t-1) + \ldots + 2t^{h-1}(t-1) \tag{1}$$
$$n \geq 1 + 2(t-1)(1 + t + t^2 + \ldots + t^{h-1}) \tag{2}$$
$$n \geq 1 + 2(t-1)\frac{t^h - 1}{t-1} \tag{3}$$
$$n \geq 1 + 2(t^h - 1) \tag{4}$$
$$n \geq 2t^h - 1 \tag{5}$$
$$t^h \leq \frac{n+1}{2} \tag{6}$$
$$h \leq \log_t \frac{n+1}{2} \tag{7}$$

**Corollary:** Height is $O(\log_t n) = O(\log n)$ when $t$ is constant.