Btree of order m (m = maximum
number of children)

A Btree of order m is a
m-way search tree where
all failures (ϕ subtrees)
are on same level. A
Btree node has structure

$t_0 \; (\boxed{K_1}\;\boxed{t_1}) \; (\boxed{K_2}\; t_2) \cdots (K_m t_m)$

① value of all keys in t_i

$$K_i < \text{all keys in } t_i \leq k_{i+1}$$

② $K_i \leq K_{i+1}$ for all
$i = 1 \ldots m$

Btree of order m = 4
(2, 4) tree

$(25, \phi)$

$m = \frac{4}{2} = 2$

old   new

(root)20   new

old
⌀ 10 20 25 30

(k,t) 25 30

t_0 $(k_1 t_1)(k_2 t_2)(k_3 t_3)$
⌀ (10 ⌀) (20 ⌀) (30 ⌀)   20

10   25 30

root →

```
class BTnode {
    public:
        const int m = 4;
        BTnode * t[m];
        int key[m-1];
}
```

void Search (int v, BTnode* p, int pos)
{

p = root;
int idx = binarySearch (p→key, v)

20

< ==   < OR >

0   1   2   3
10   20   30

0   1   2   3
10   20   30

---

20   19

10 15 18   19   25 30

20
15
⌀ 10 ⌀   ⌀ 18 ⌀ 19 ⌀   25 30

Remove
Keys

$m = 300$

20

17

10 15

18

28 30

root

17 20

10 15

18

25 30