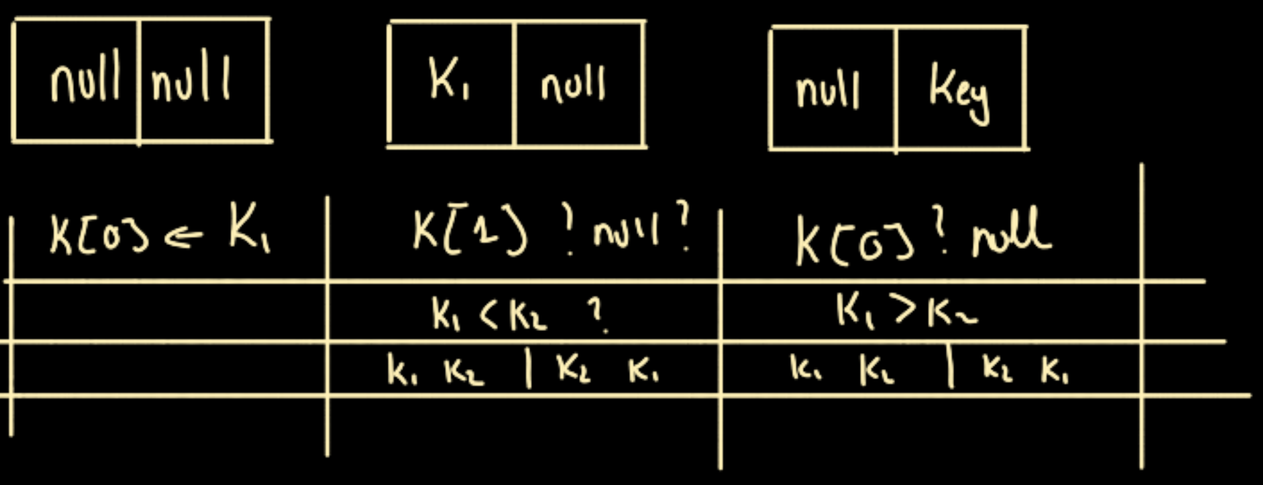
B3R-Tree

→ Node Structure

1. Representation :

Attributes
- Pointers → Parent Node, L, C, R Child Nodes
- Keys ← K_1 & K_2
- Leaf

2. Insert Keys :




```
Insert (K)
if (no Space) → exit
if  K1 OR K2 == null
|    if K1 == null OR K2 < K
|    |    if K1 != null
|    |    |  K2 = K1
|    |    K1 = K
|    else
|    |  K2 = K
else if  K < K1
|    K2 = K1
|    K1 = K
else if  K < K2
|    K2 = K
```

3. Remove

```
removeKey (K)
for i in Keys.length
|    if Key[i] == K
|    |    Key[i] ⇐ null
|    |    exit
```

IF KEY NEEDED →

```
popKey (K)
for i in Keys.length
|    if Key[i] == K && != null
|    |    pop ← Key[i]
|    |    Key[i] ⇐ null
|    |    exit
return null;
```