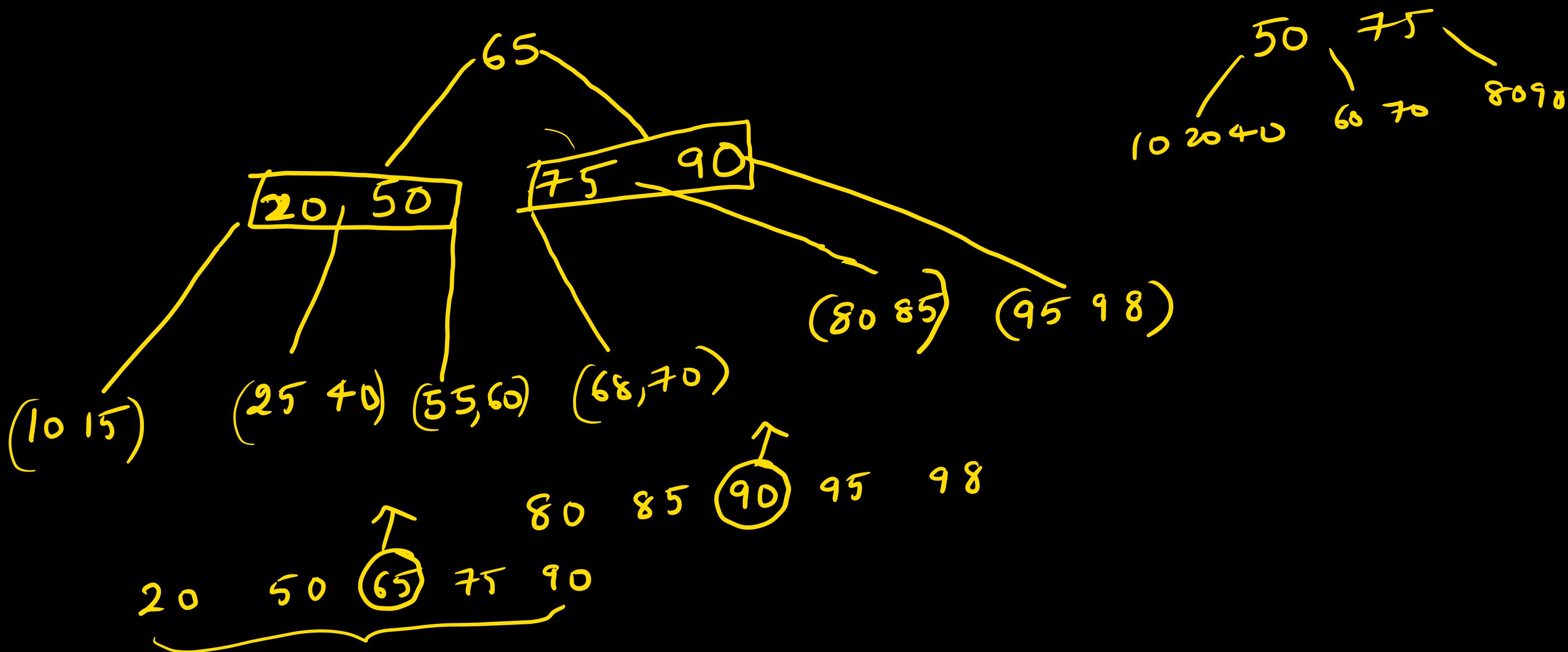


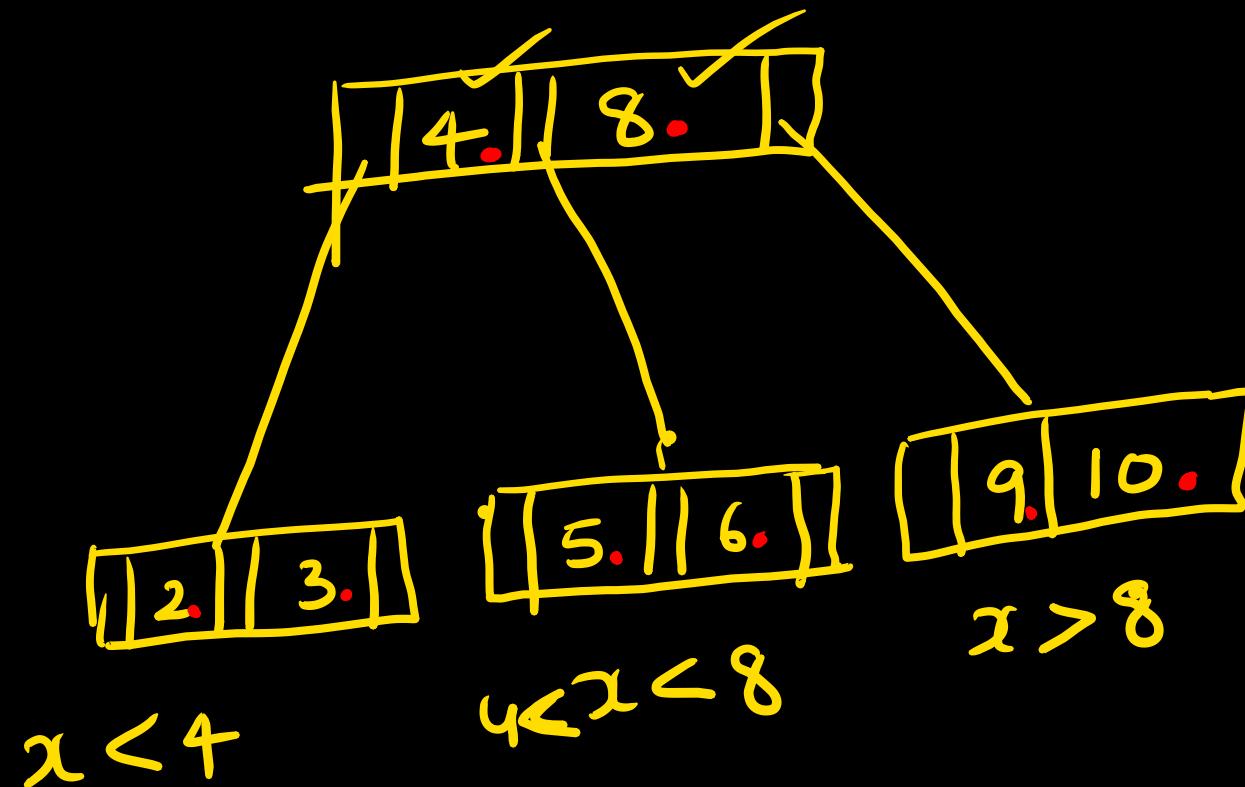
60 40, 60, 20, 50) 70, 90, 80, 75, 10, 15, 25, 55, 65, 68, 85 98 95
↑ ↑



In most cases, p is ~~not~~ max no of block pointers. But
this definition may change in the questions. Then go with
q data given.

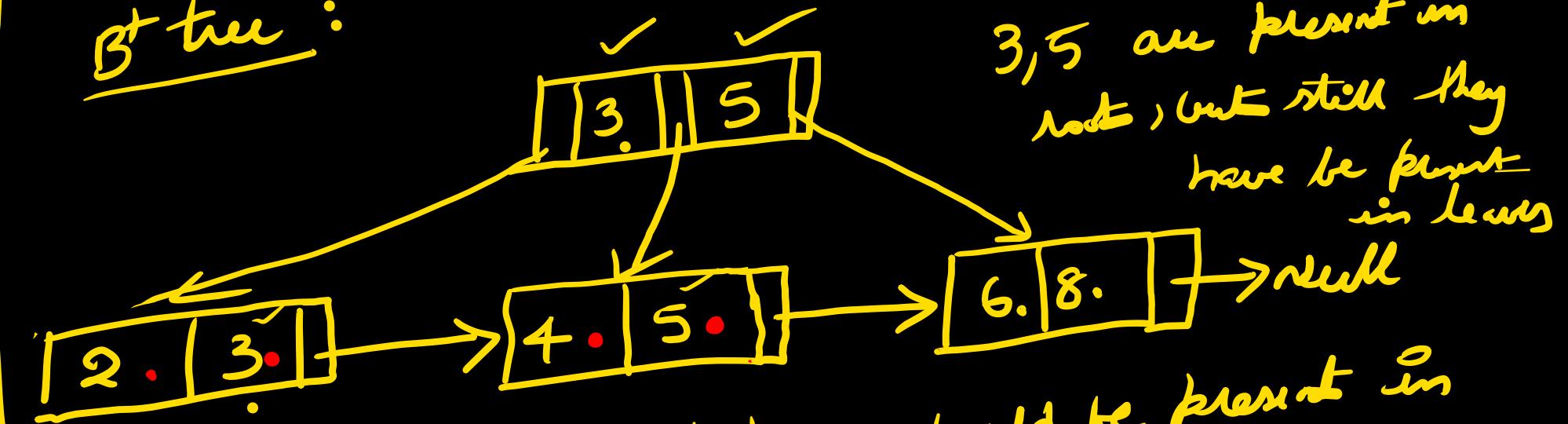
For x : Order P : ~~Max~~ Keys in a node.

Btree



4, 8 are present in non leaf
so they need not be present
in leaf node

B⁺ tree :



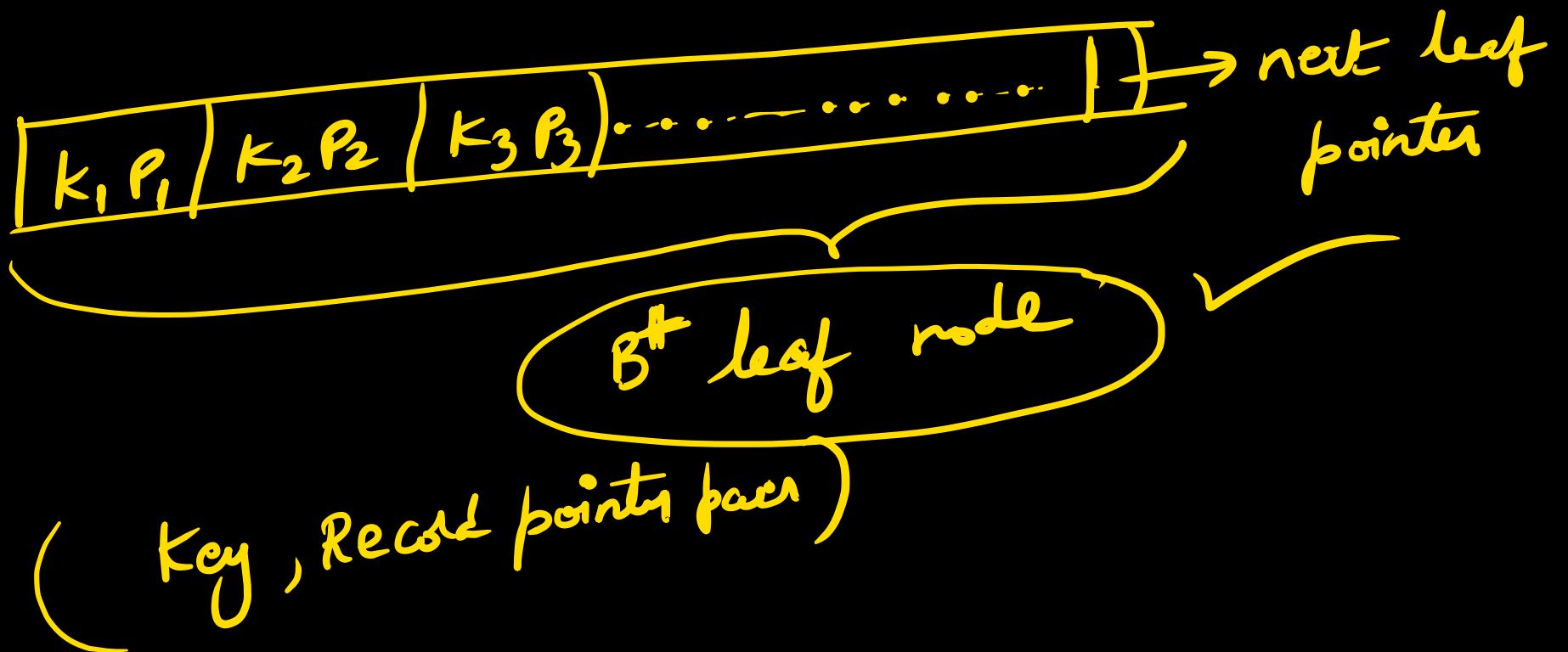
all the keys in B⁺ tree should be present in leaves.

In B⁺ tree data pointer/recd pointer is present only in leaves. In B tree it is present in all the nodes

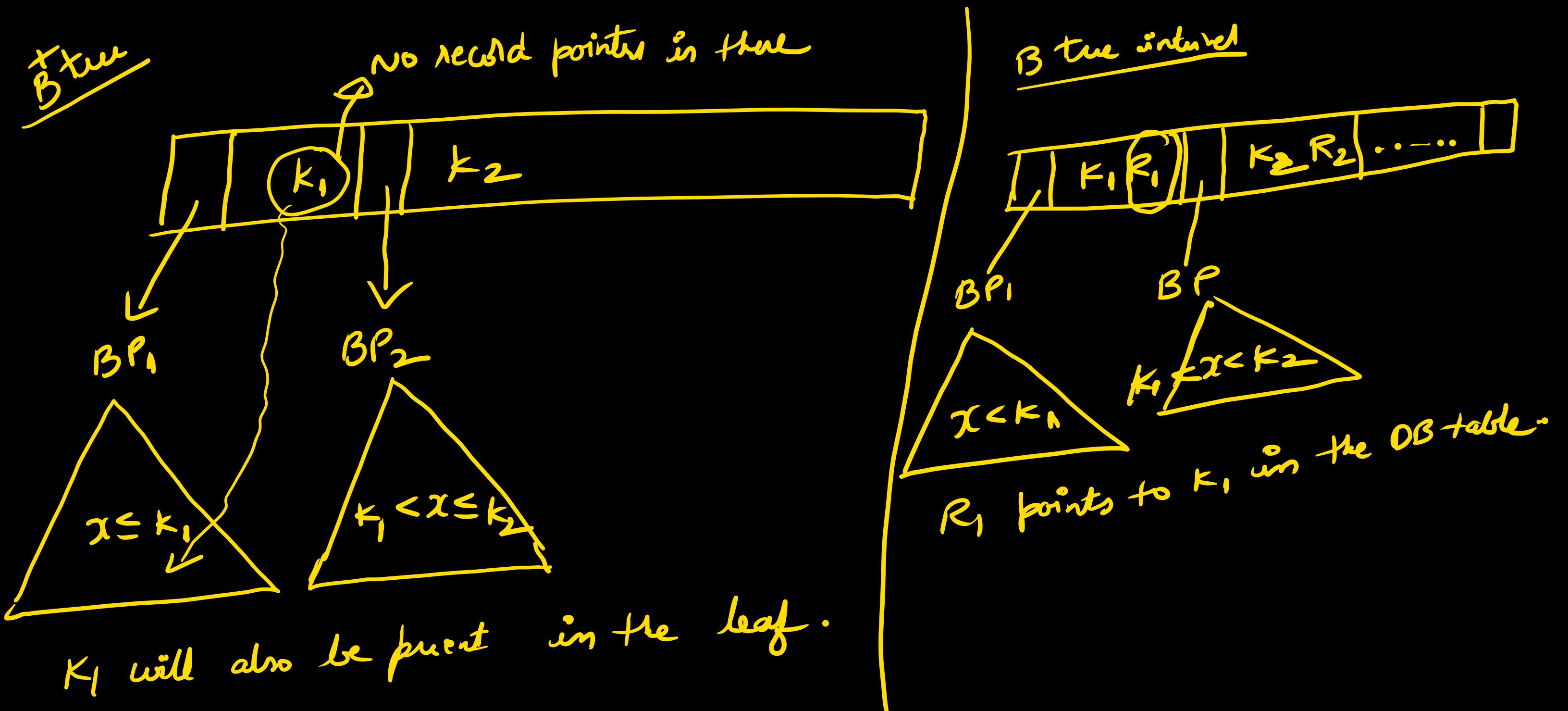
not leaf ptr is present in every leaf... for range queries it is more suitable

3, 5 are present in root, but still they have to be present in leaves

Node structure :



a is a pointer to 1, b is a pointer to 2, ...



Insert 60, 40, 20, 50, 30 into B^t tree of Order 5 • (maximum number pointers per node)
 (max number of keys 4.)



leaf node

④ RP ① block pointer.

P=5

4 RP + 1 BP

keys : ④ ✓

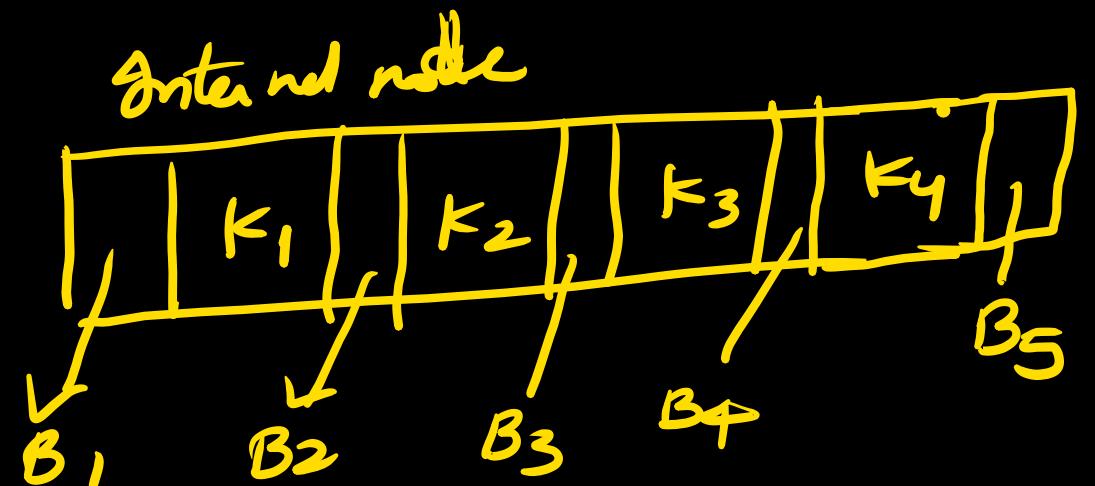
leaf &

max t+1=5

granted :

key are same

max no of pointers is also same.



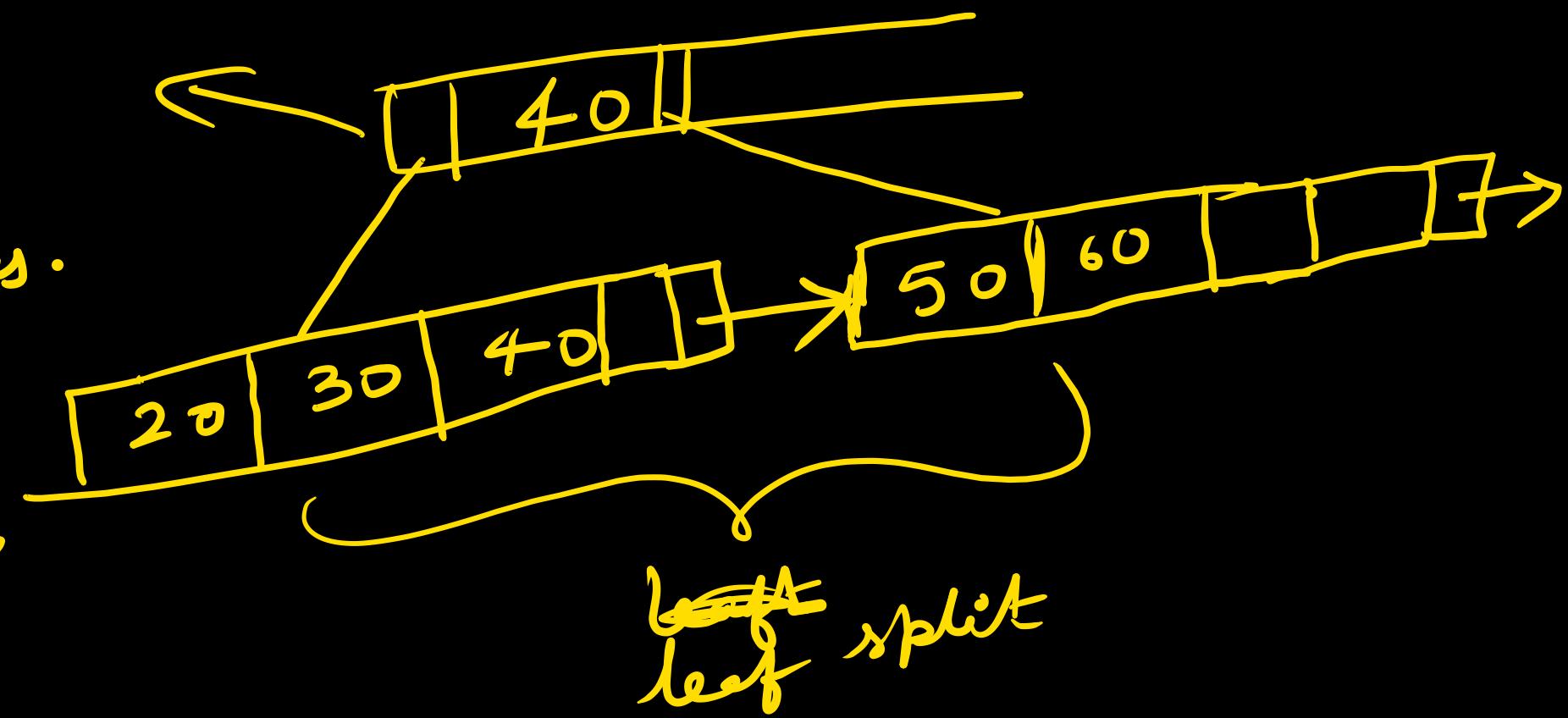
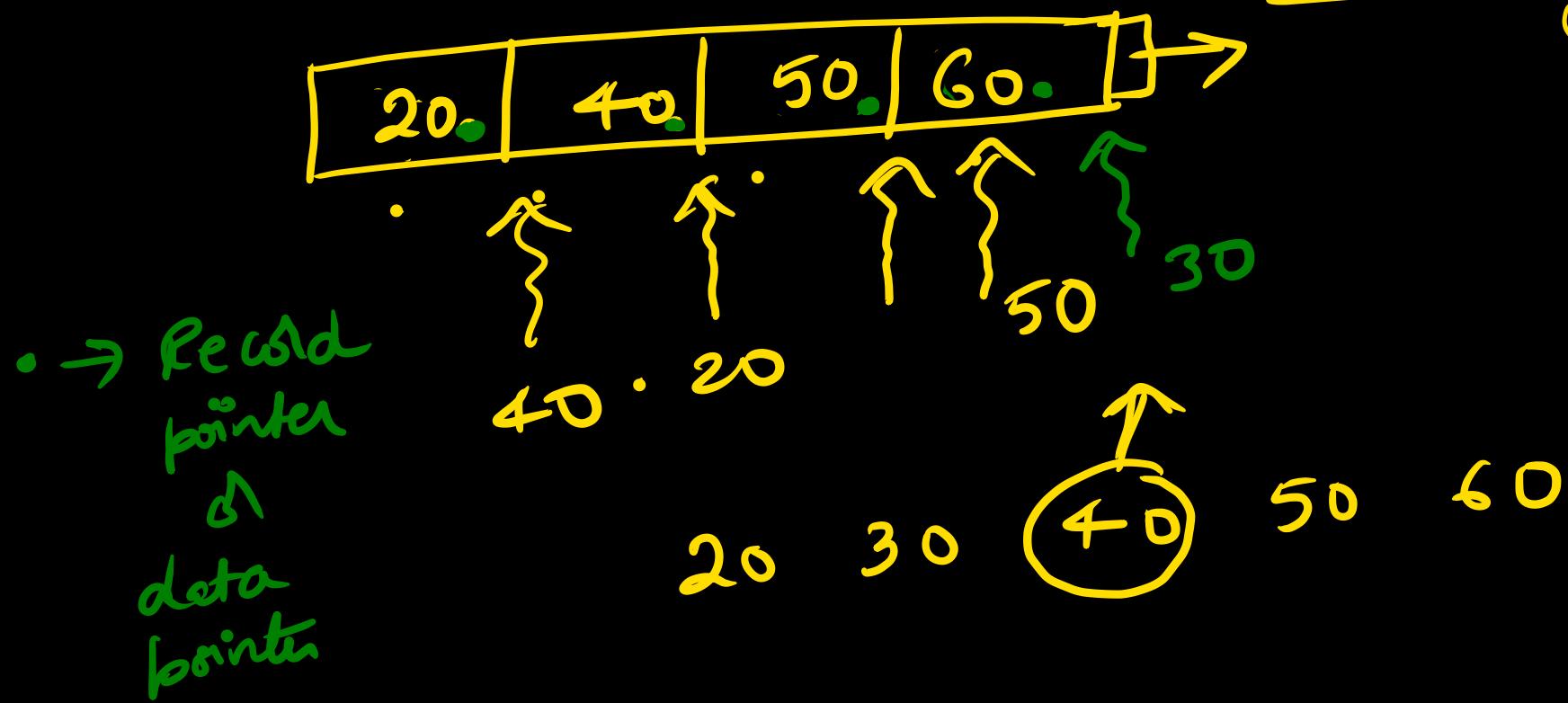
internal node

BP = 5 ✓

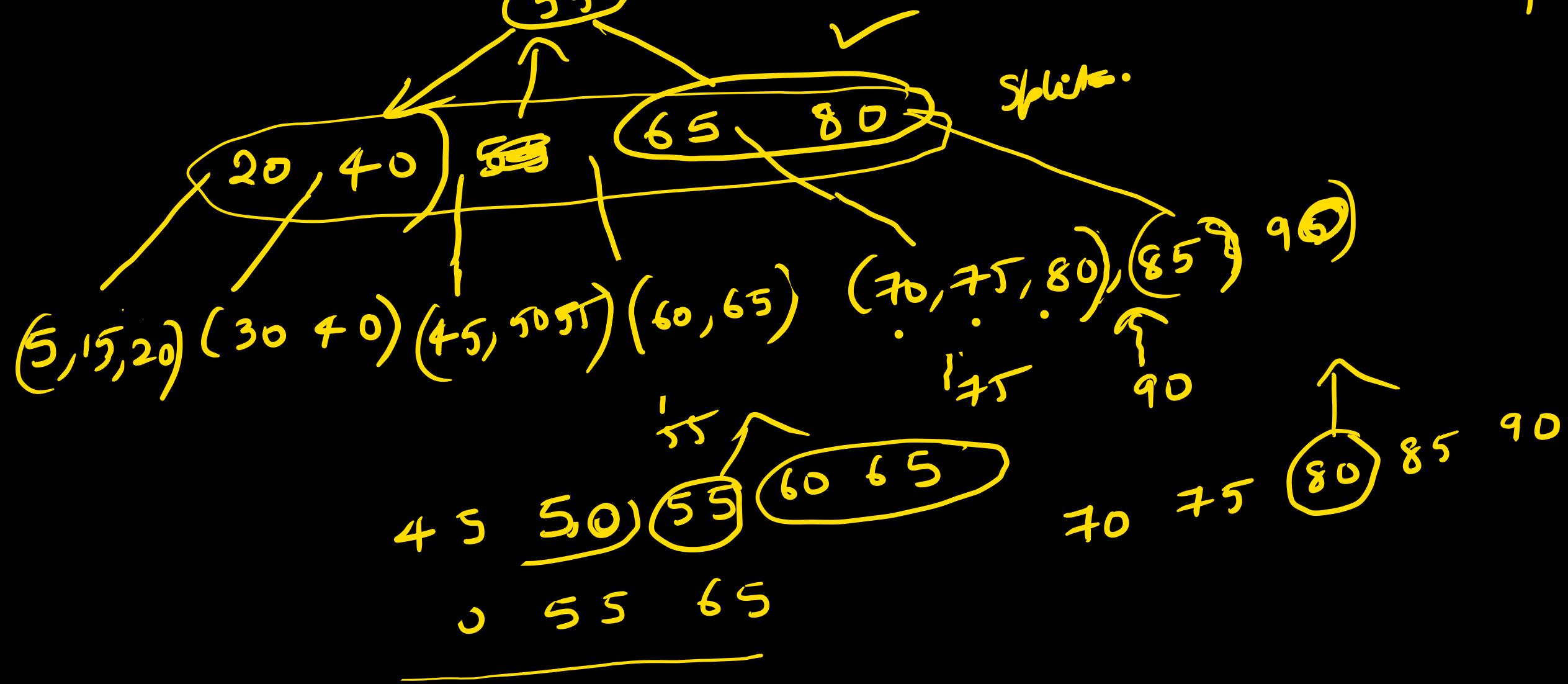
keys = 4.

given 60, 40, 20, 50, 30

Splitting non leaf is
exactly like B trees.



60 40 20 50 30 80 70 65, 15, 5, 45, 55 75 85, 90



splitting interval node, it is same as
 B tree node.

Note: If block size = 1024, search key - 10 bytes Block pointer is
 8 bytes, record pointer is 9 bytes
 what is order P : maximum possible block pointers which can be
 stored in B tree node
 \dots

 $B P \circlearrowleft$
 \dots

$27P \leq 1043$
 $P \leq \left\lfloor \frac{1043}{27} \right\rfloor$
 $= 38.$

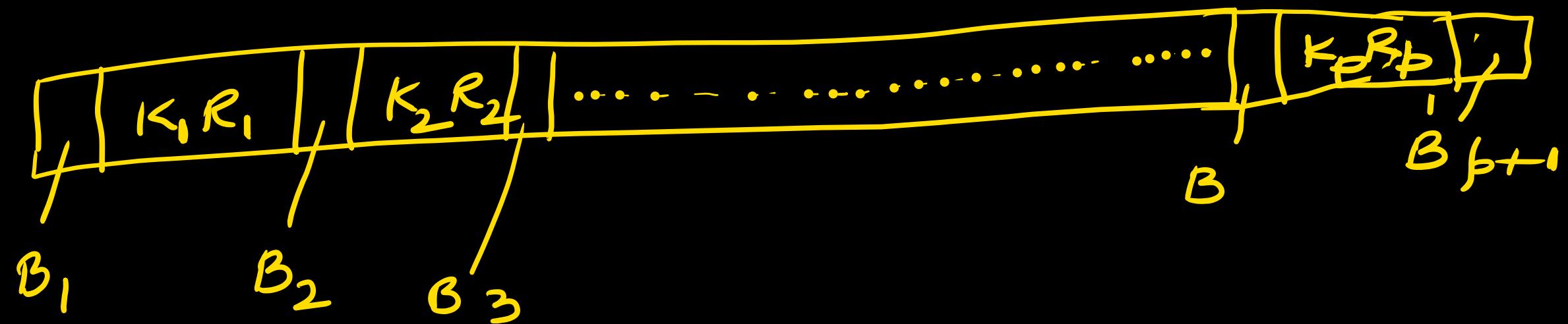
$$P * BP + \overline{(P-1)(K+RP)} \leq \text{Block Size}$$

$$\underline{P * 8 + (P-1)(10+9)} \leq 1024$$

$$\rightarrow \text{Block size} = 512 \text{ B} \quad \text{Key} = 5 \text{ bytes} \quad \underline{BP} = \underline{RP} = \underline{10B}$$

order P : max possible keys which can be stored in B tree node, what is the best possible order of B tree node

p : max possible keys



* max number of keys in P .

$$p(K+R) + (p+1)B \leq 512 \text{ B}$$

order is not always the same, it varies from question to question.

Block size is 1024, B_p is 8 bytes, R_p : 9 bytes, search key: 10 bytes

Order P: maximum pointers which can be stored in B^+ node.

What is the best possible order of B^+ tree?

- (i) Interval (ii) leaf node

Interval:



$$p(B) + (p-1)(keys) \leq \text{Block size}$$

$$p \times 8 + (p-1)10 \leq 1024.$$

$$18p \leq 1034$$

$$p \leq \lfloor 1034/18 \rfloor = 57$$

leaf node:

order P : max pointers



$$\text{Total} = \underbrace{x}_{\downarrow} \underbrace{k+1}_{\downarrow} + \underbrace{RP}_{\circled{RP}} + \underbrace{BP}_{\circled{BP}} \leq \text{Block}$$

$$(p-1)(k+RP) + BP \leq \text{Block}$$

$$(p-1)(10+9) + 8 \leq 1024$$

$$19P \leq 1035$$

$$P \leq \left[\frac{1035}{19} \right] - 54$$

$$x = \underbrace{p-1}_{\circled{p-1}} \quad \checkmark$$