

INTRODUCTION TO

B+ TREES

DEFINITION

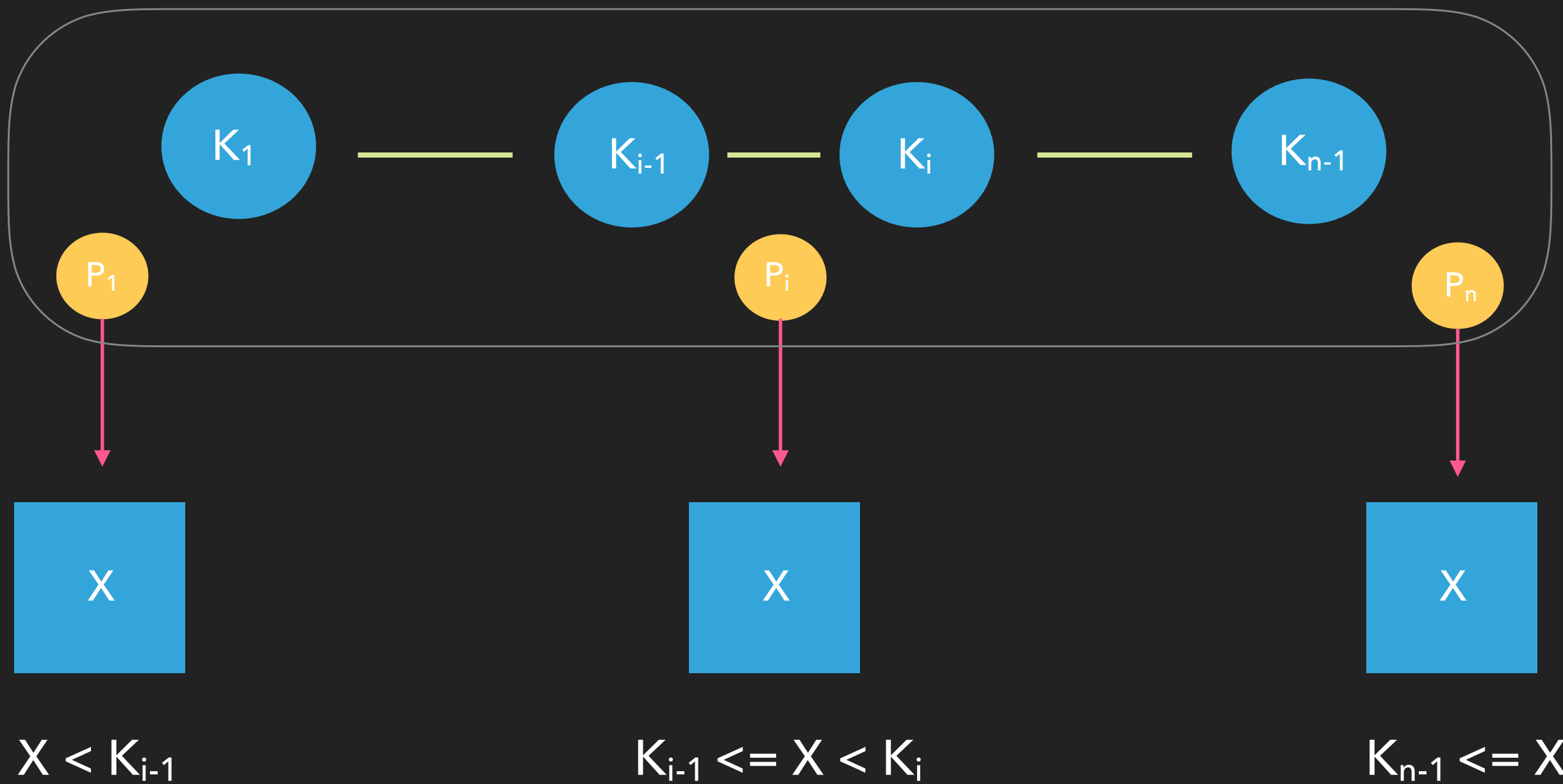
- ▶ It is the combination of ISAM (Indexed Sequential Access Method) and B tree.
- ▶ Entire input data is present at the leaf nodes.
- ▶ Every leaf node is interlinked with each other.
- ▶ Leaf node has a data pointer which points to the data present in the hard disk.
- ▶ B+ tree has two orders:
 1. For the internal nodes.
 2. For the external (or leaf) nodes.

DESIGN OF INTERNAL NODE

- ▶ Each internal node is of the form:
$$\langle P_1, K_1, P_2, K_2 \dots P_{n-1}, K_{n-1}, P_n \rangle$$

where $n \leq a$, P_i points to another node of the tree,
 K_i is a key value.
- ▶ Every node has $K_1 < K_2 < K_3 \dots < K_{n-1}$
- ▶ Field value X of sub-tree pointed by P_i holds the following condition:
$$K_{i-1} \leq X < K_i \quad (1 < i < n)$$
- ▶ $K_{i-1} \leq X \quad (i = n)$

DESIGN OF INTERNAL NODES

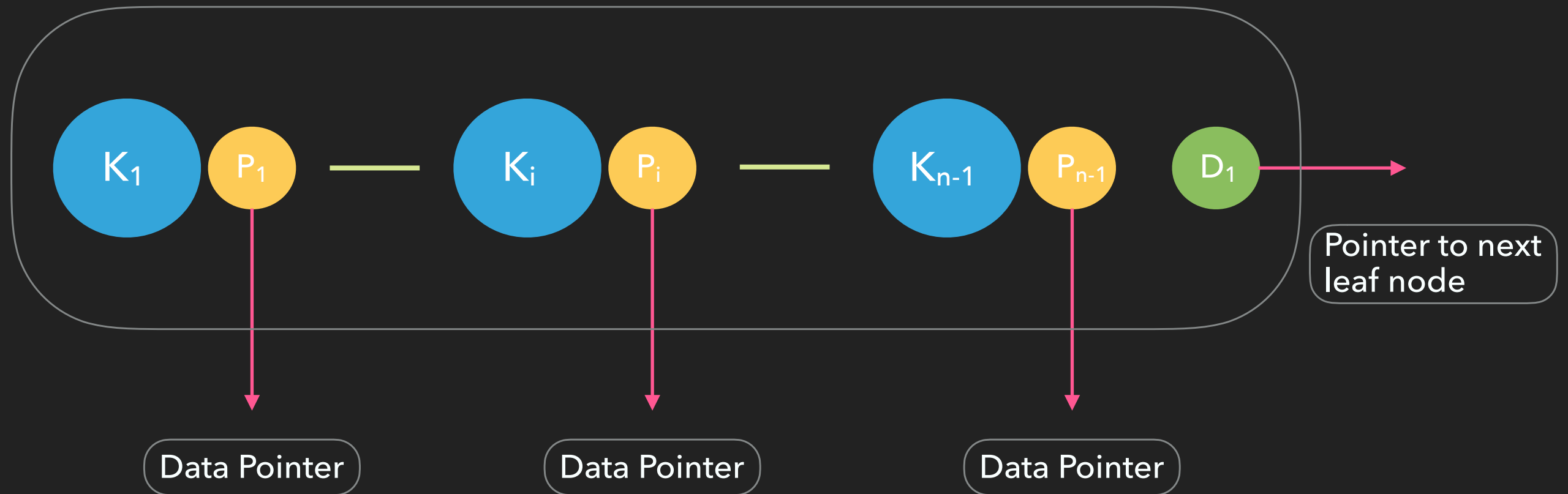


DESIGN OF LEAF NODE

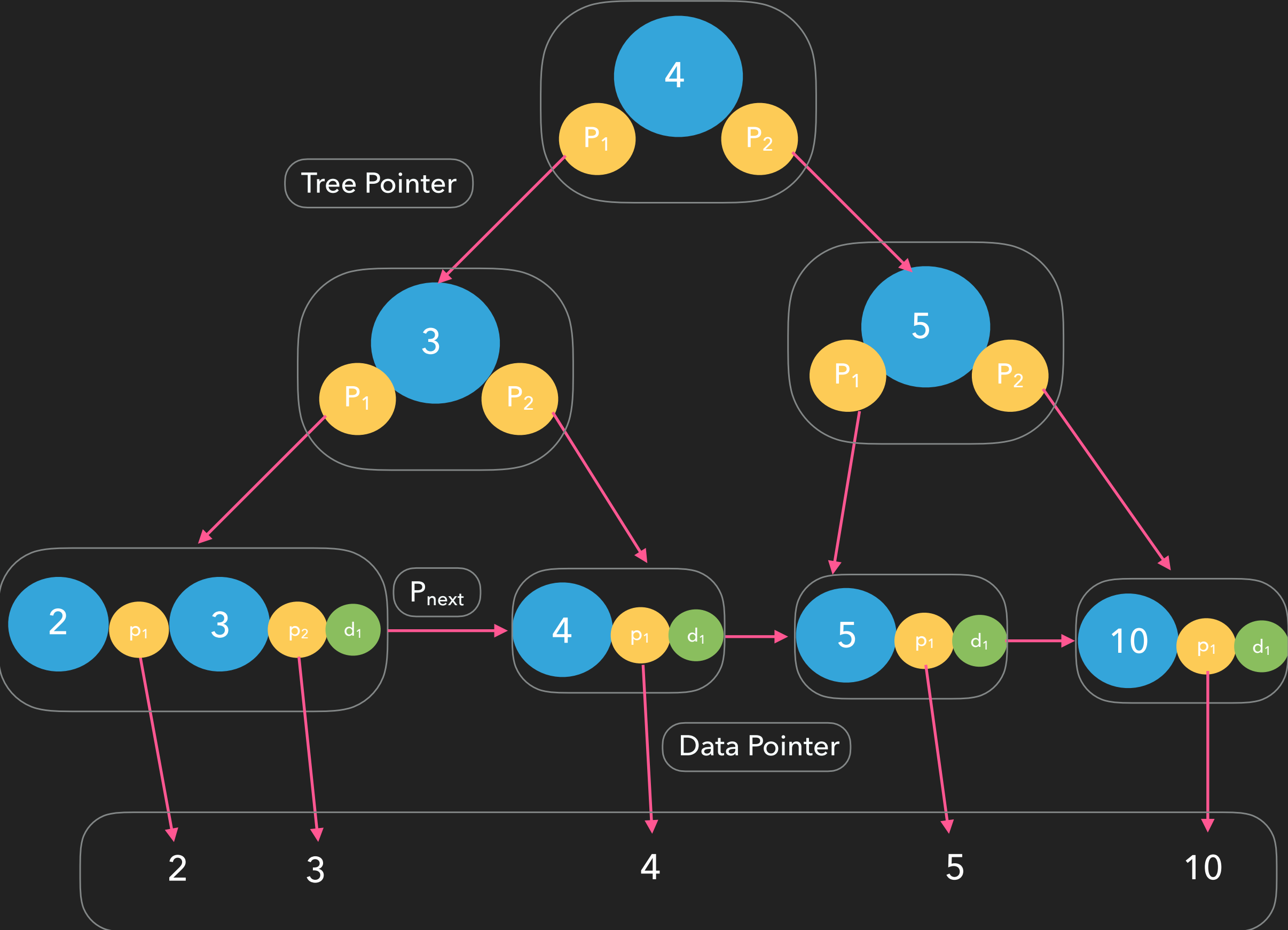
- ▶ Each external node is of the form:
$$\langle \langle K_1, D_1 \rangle \langle K_2, D_2 \rangle \dots \langle K_{n-1}, D_{n-1} \rangle, P_{\text{next}} \rangle$$

where $n \leq b$, D_i points to actual record in the disk whose key value K_i or to a disk file record containing the record .
- ▶ Every node has $K_1 < K_2 < K_3 \dots < K_{n-1}$ ($n \leq b$).
- ▶ Each leaf node has at least $\lceil b/2 \rceil$ values.
- ▶ All leaf nodes are at the same level.

DESIGN OF LEAF NODES



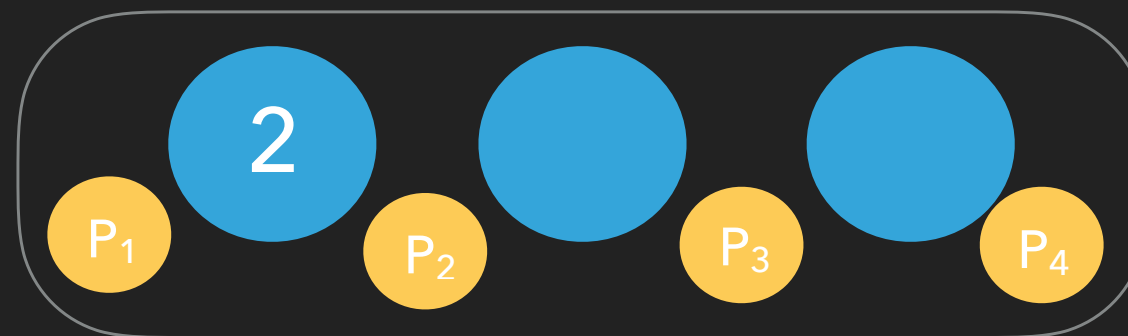
DESIGN OF A B+ TREE



Assuming order of internal node is: n

INSERTION

2-> 4-> 7-> 10-> 17-> 21-> 28->



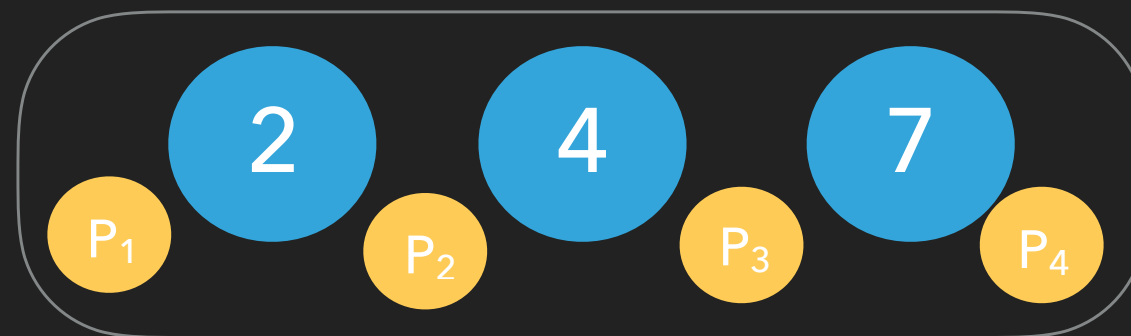
INSERTION

4-> 7-> 10-> 17-> 21-> 28->



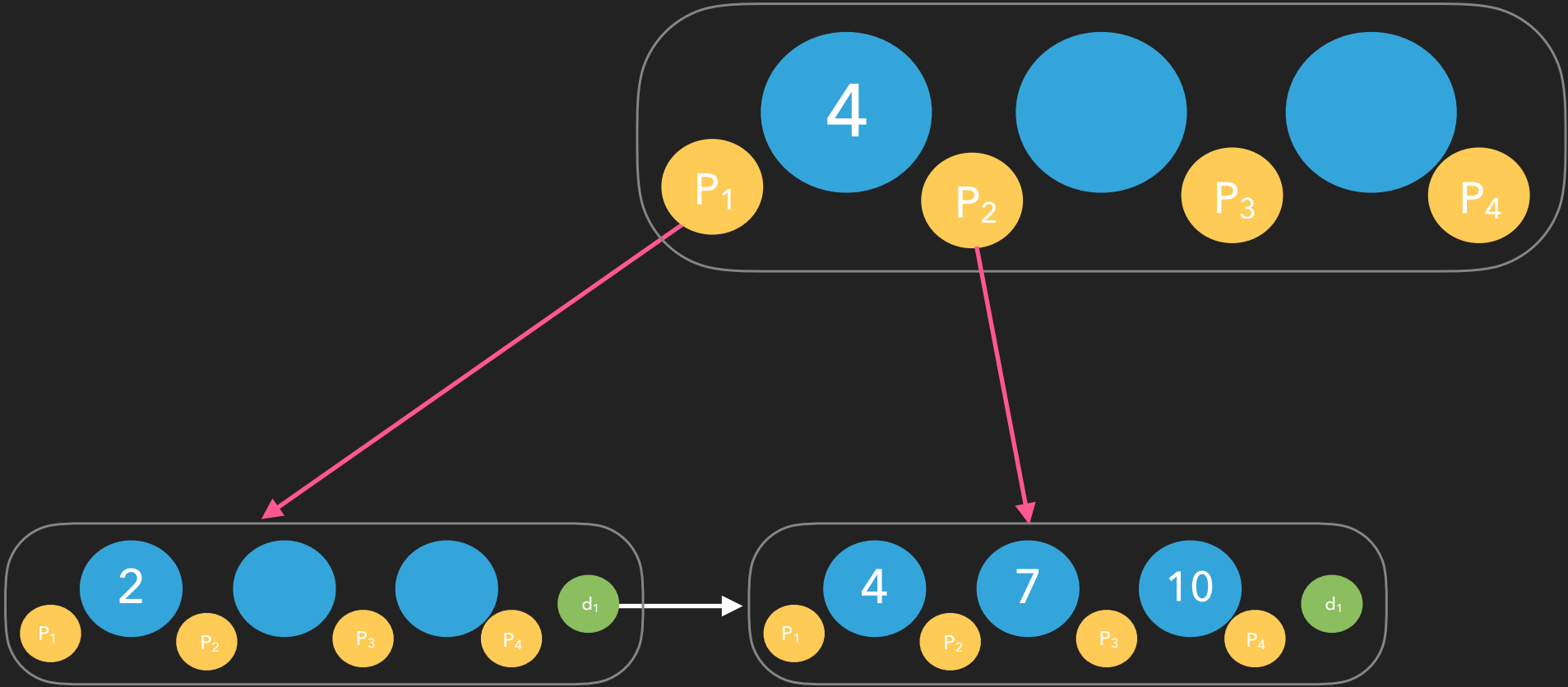
INSERTION

7-> 10-> 17-> 21-> 28->



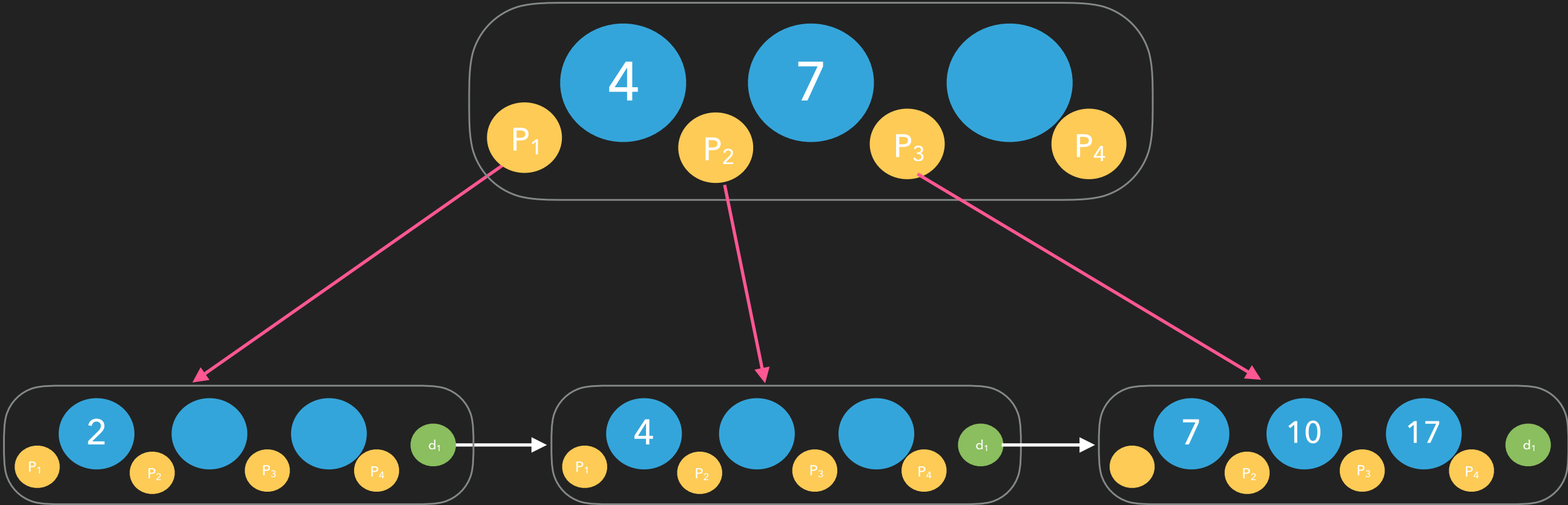
INSERTION

10-> 17-> 21-> 28->



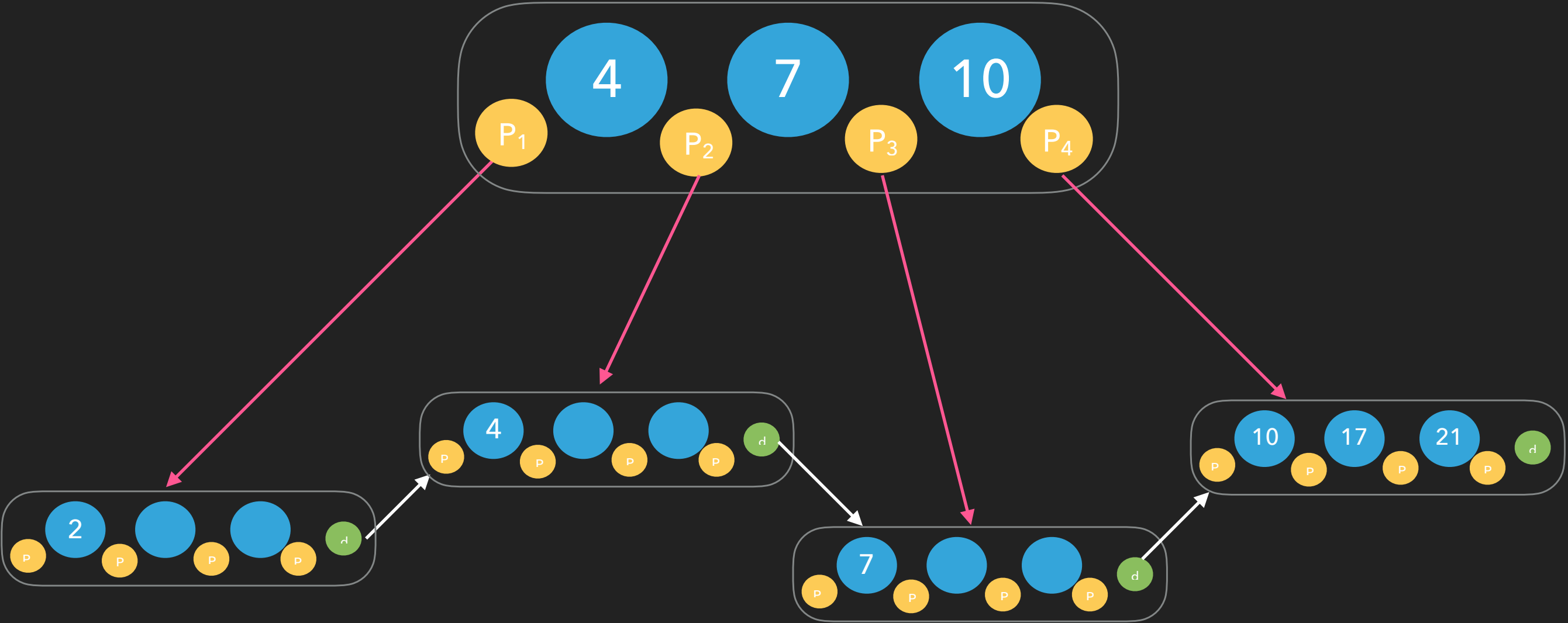
INSERTION

17-> 21-> 28->



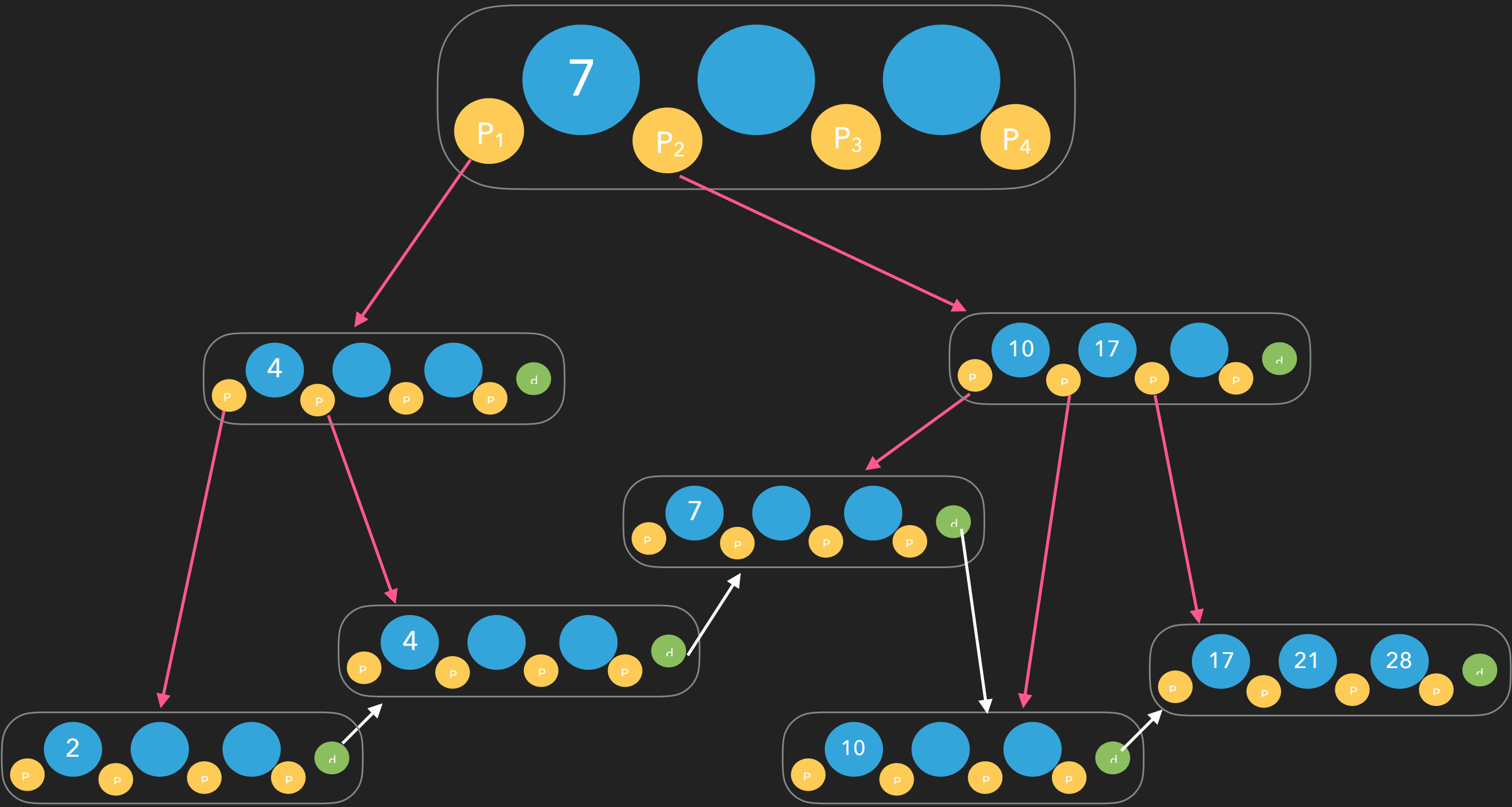
INSERTION

21-> 28->



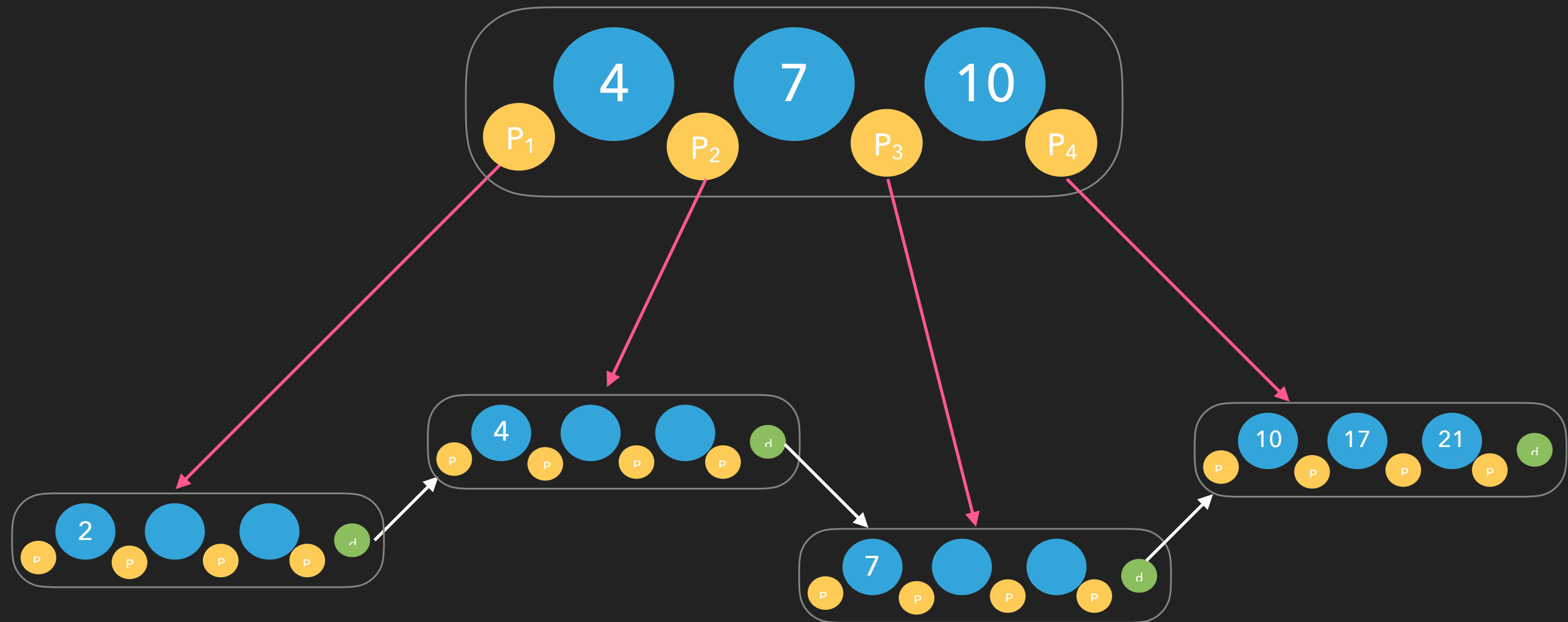
INSERTION

28->



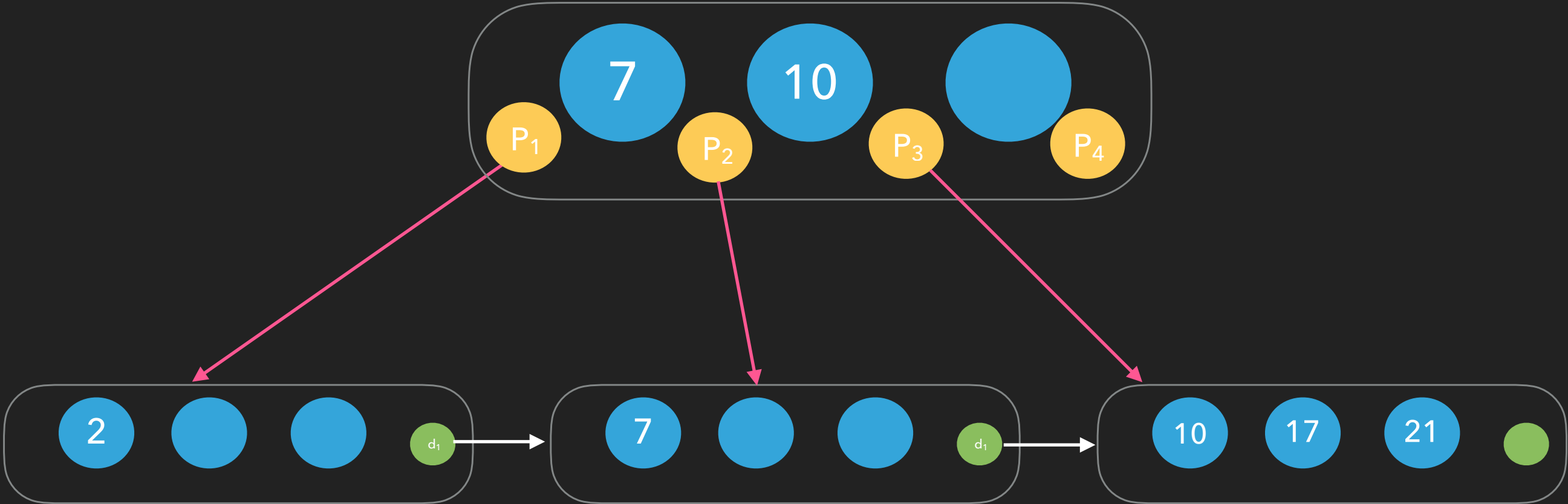
DELETION

28->



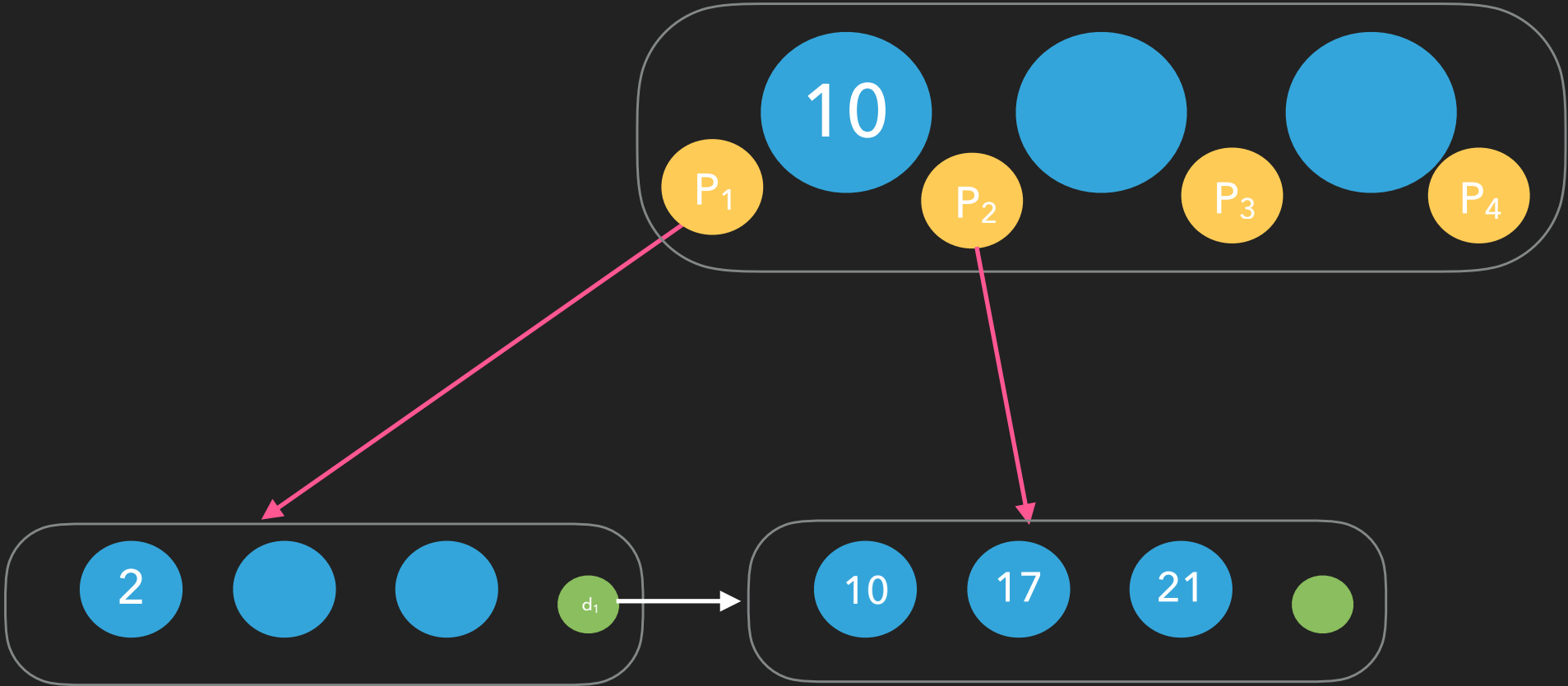
DELETION

4->



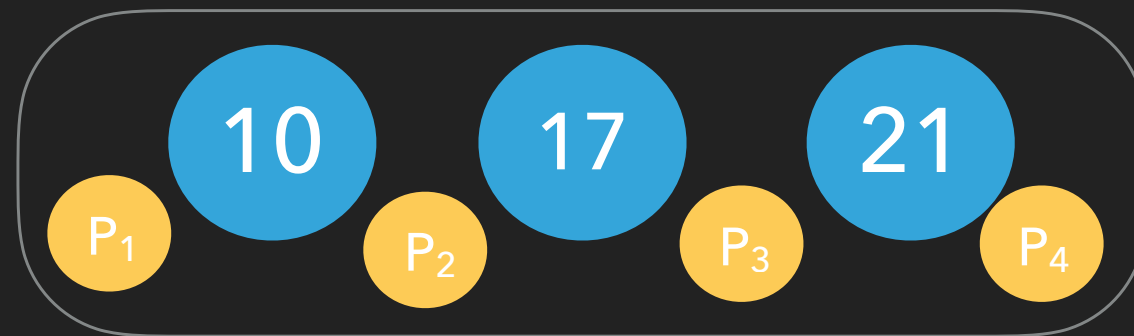
DELETION

7->



DELETION

2->



ADVANTAGES OVER B TREE

- ▶ The leaf nodes of B+ trees are linked, so a full scan in order traversal requires just a linear pass through all leaf nodes.
- ▶ B+ trees can fit lots of data in the internal data as no space is required for the data recorded in the leaves. This leads to lower tree height and better cache utilisation as more keys can fit on a page of memory, hence cache misses are reduced.
- ▶ Deletion doesn't propagate as it is performed only at the leaf nodes.