

CS3402 Tutorial 6:

Considering a B+ tree for indexing. Each node in the tree can fit four pointers and three key values, denoted as $\langle A_1, K_1, A_2, K_2, A_3, K_3, A_4 \rangle$. Within each node, the keys satisfy $K_1 < K_2 < K_3$. For all search field values X in the sub-tree pointed by A_i , we have:

$$\begin{aligned} K_{i-1} &\leq X < K_i \text{ for } 1 < i < 4, \\ X &< K_i \text{ for } i=1, \\ K_{i-1} &\leq X \text{ for } i = 4. \end{aligned}$$

1. Construct a B+ tree for the following set of key values:

(2, 3, 5, 7, 11, 17, 19, 23, 29, 31)

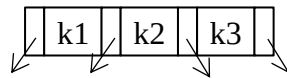
Assuming that the tree is initially empty, values are added in ascending order, and the number of key values in internal nodes and leaf nodes are both 3.

2. For the B+ tree constructed for question 1, show the form of the tree after each of the following series of operations:
 - (a) Insert 9
 - (b) Insert 10
 - (c) Insert 8
 - (d) Delete 7
 - (e) Delete 8
 - (f) Delete 5
 - (g) Delete 19

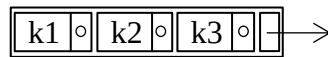
CS3402 Tutorial 6:

1. **Answer:**

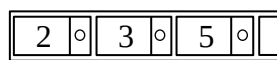
- When the number of key value in internal nodes is 3, a full internal node of this B+ tree will look like:



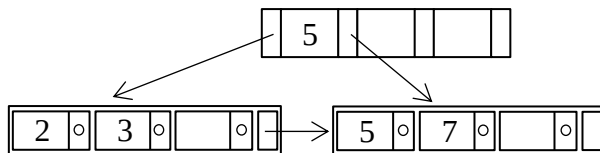
- When the number of key value in leaf nodes is 3, a full leaf node of this B+ tree will look like:



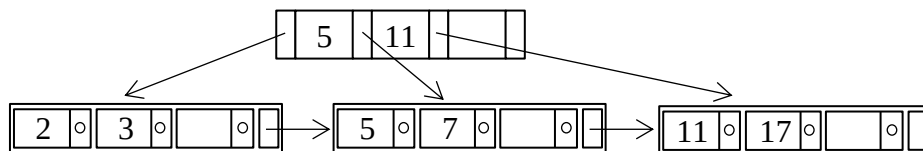
- After inserting 2, 3, 5, the tree looks like



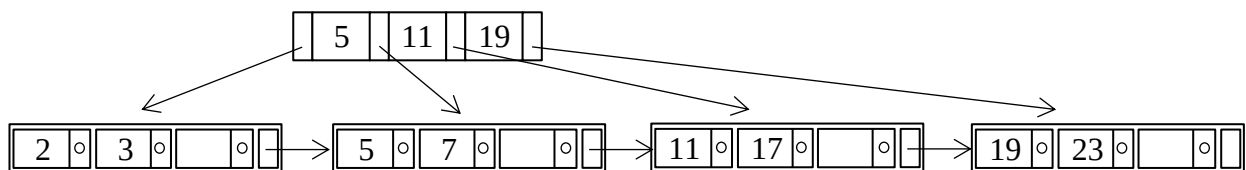
- After inserting 7, the tree looks like



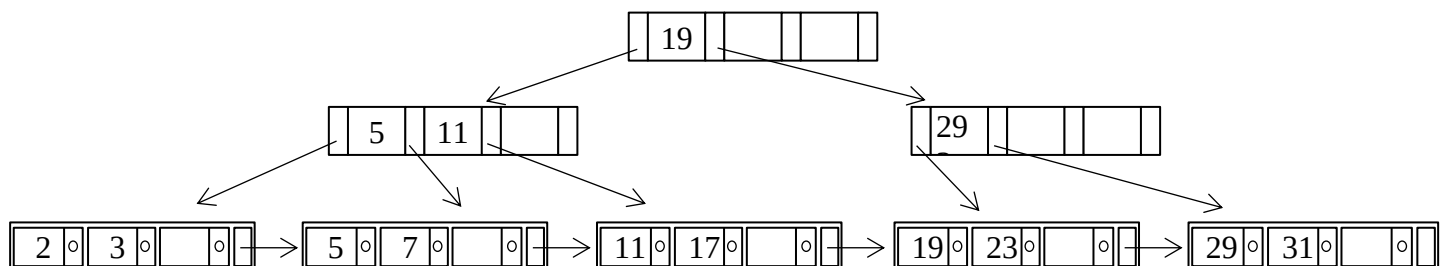
- After inserting 11, 17, the tree looks like



- After inserting 19, 23, the tree looks like

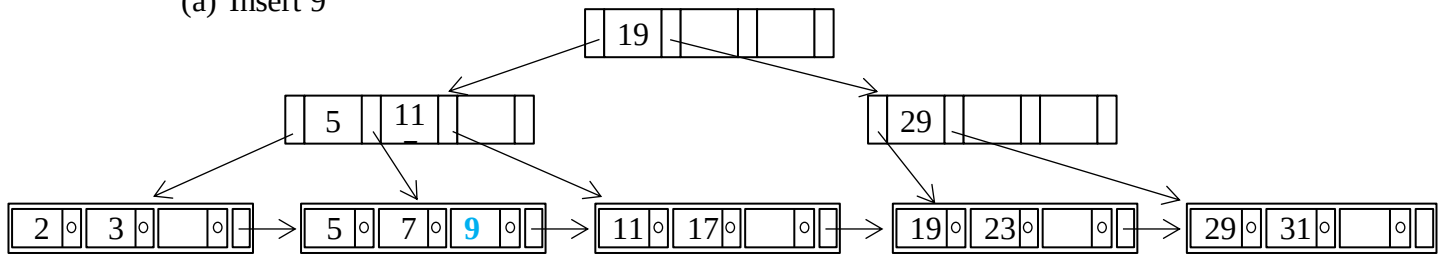


- After inserting 29, 31, the tree looks like

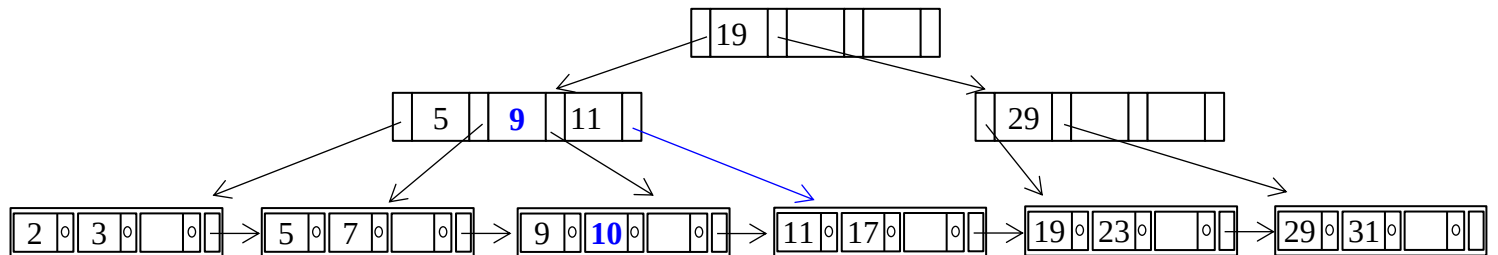


2. Answer:

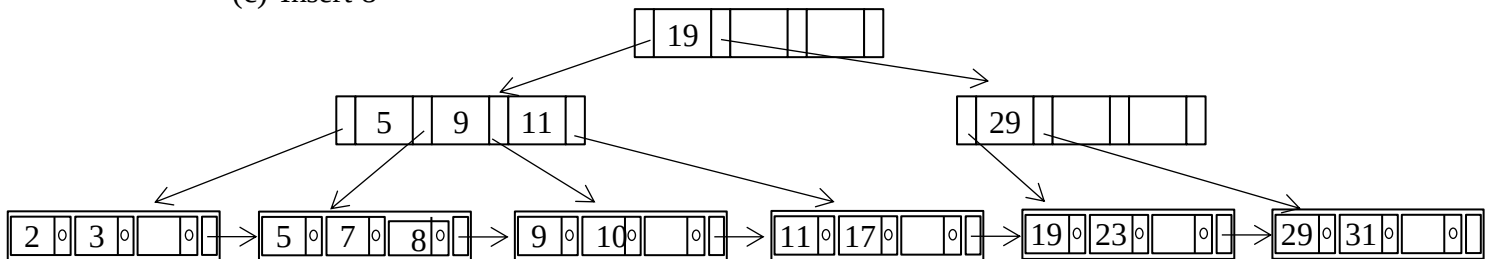
(a) Insert 9



(b) Insert 10

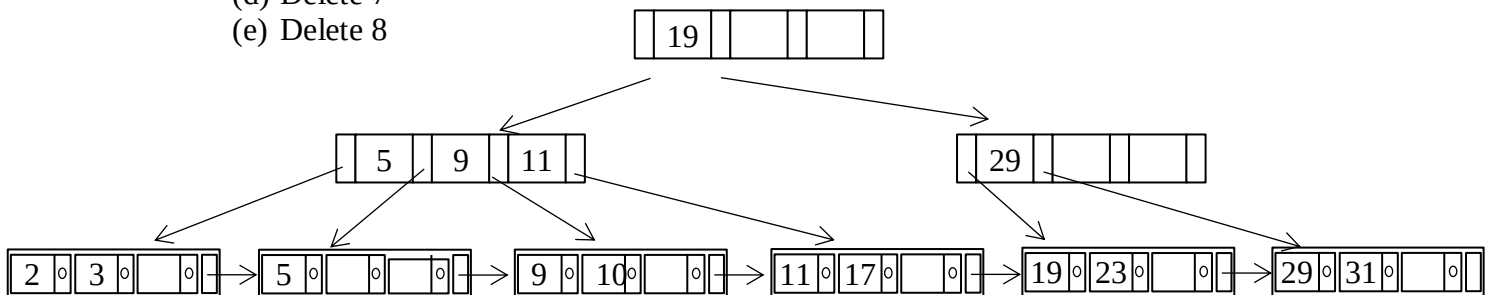


(c) Insert 8

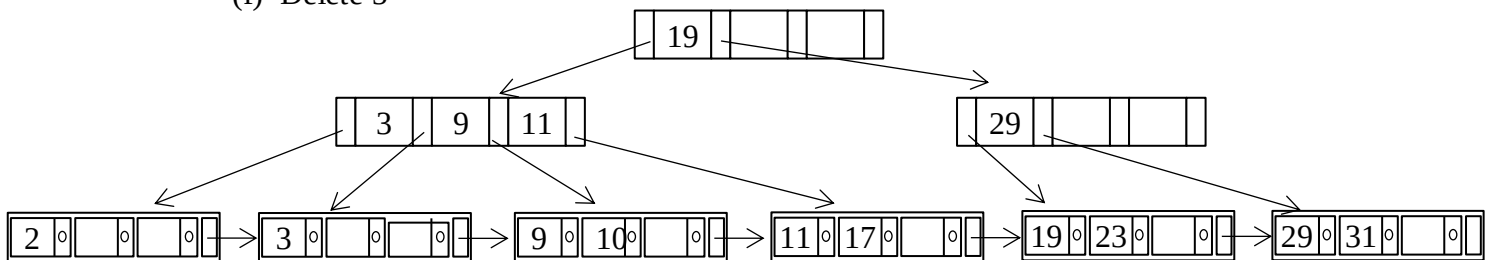


(d) Delete 7

(e) Delete 8



(f) Delete 5



(g) Delete 19

