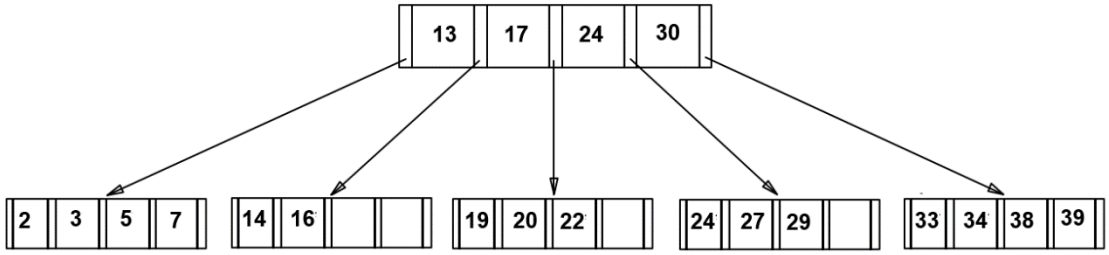
Tutorial – 11: Indexing & Hashing

1) Consider the below B+ tree



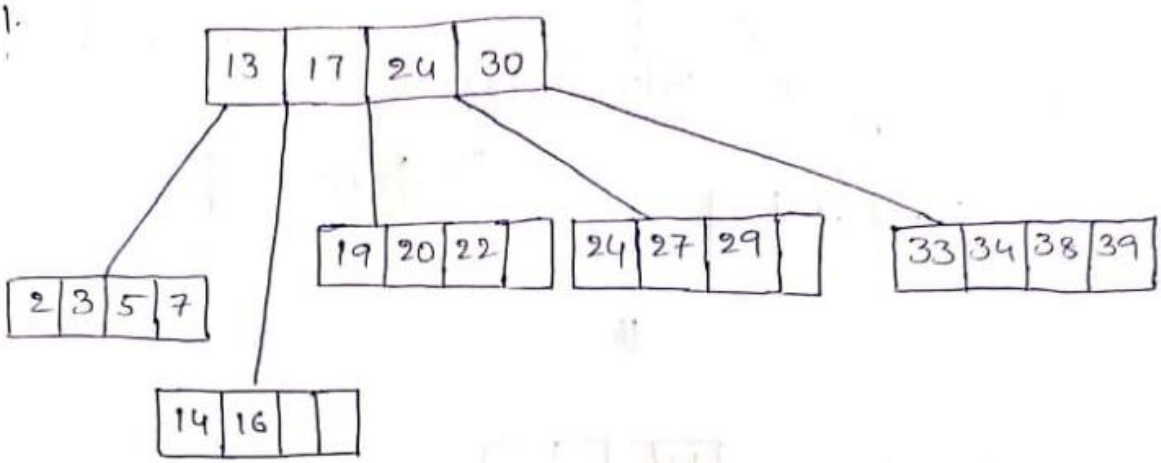
Perform the following operations on the above tree in the order as given below where the upper limit to the degree (no. of children or pointers) of the tree is 5 (i.e. no. of search key values in a node that can be allowed <= 4). During insertion, assume that splitting operation involves placing first $\lfloor n/2 \rfloor$ in the original node. During deletion, assume that redistribution operation is the preferred choice over merge siblings operation:

a) Insert 23
b) Insert 8
c) Delete 19
d) Delete 20
e) Delete 24

Tutorial - 11

190031187
Radhakrishna

1.



The above b+ tree is of order 5

i.e $m = 5$

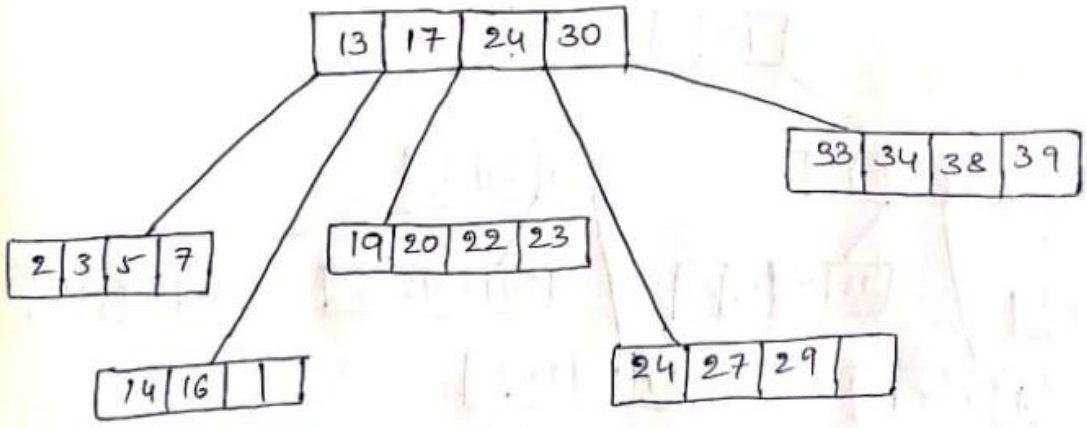
Therefore, maximum children = 5

minimum children = $5/2 = 3$

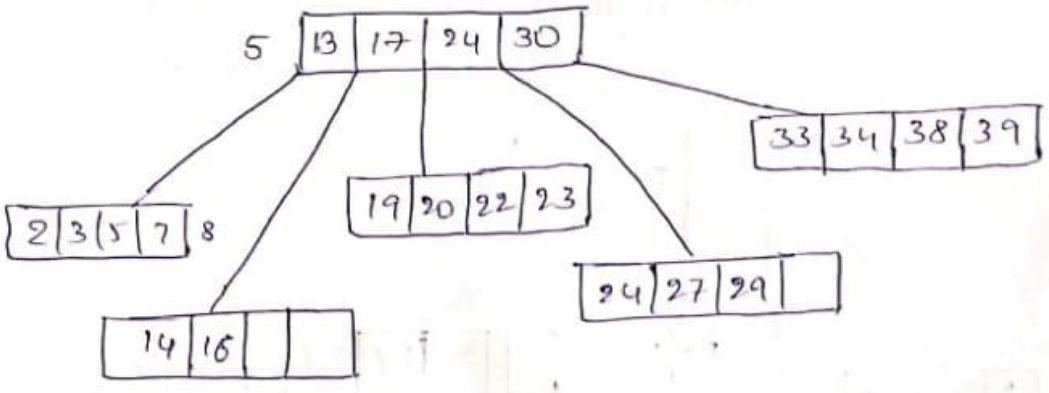
maximum keys = 4

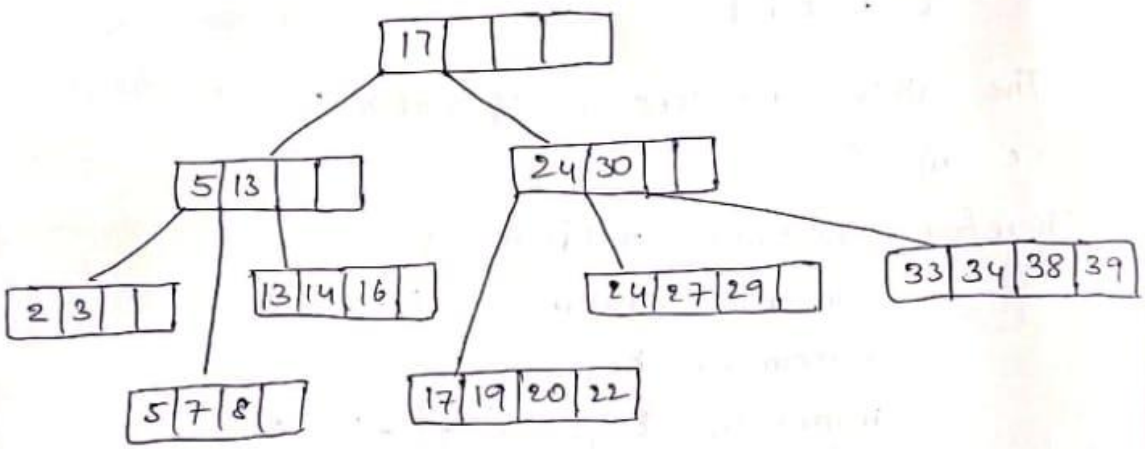
minimum keys = $\lceil 5/2 \rceil - 1 = 2$

a) Insert 23

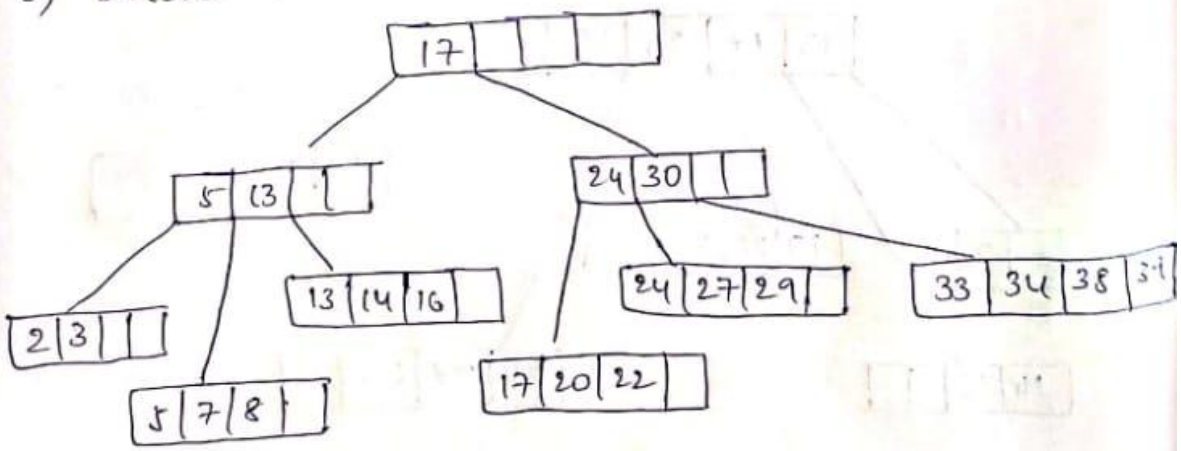


b) Insert 8

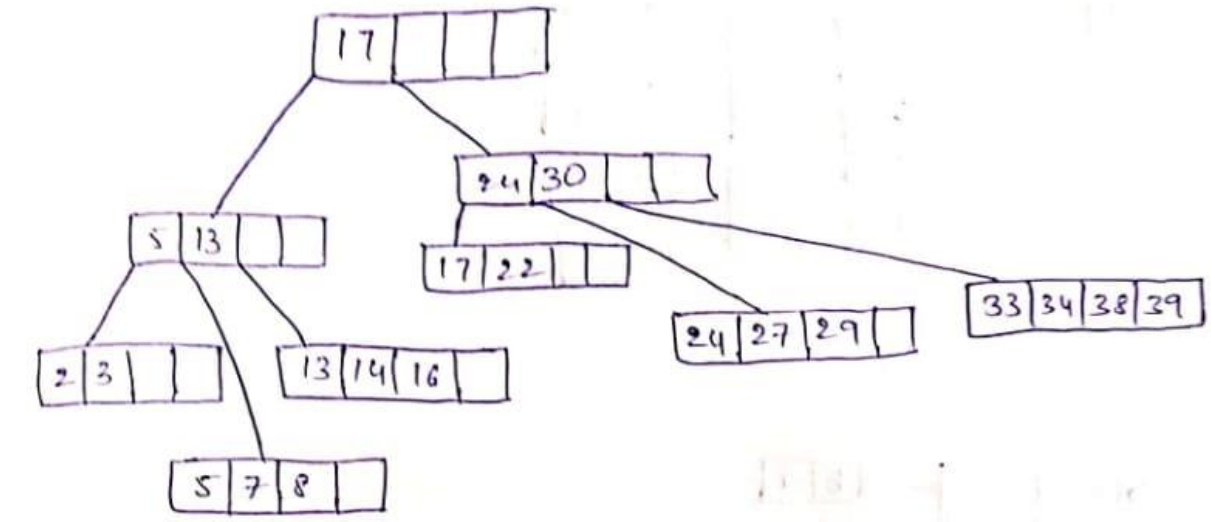




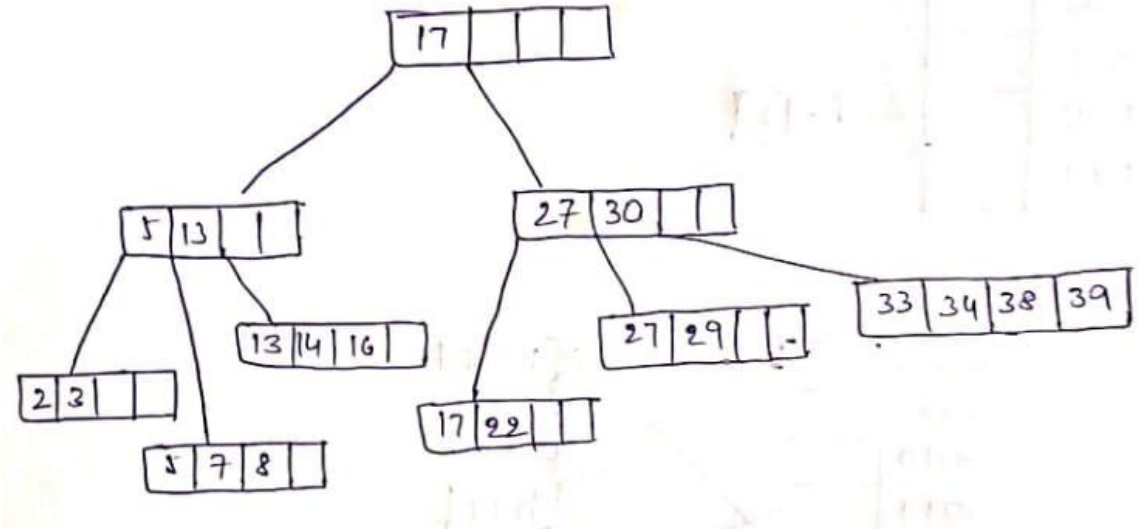
c) Delete 19



d) delete 20

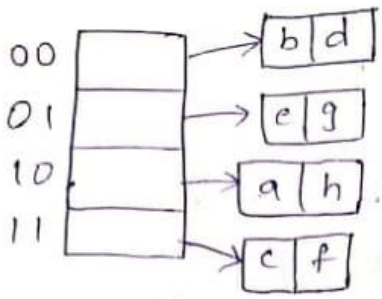


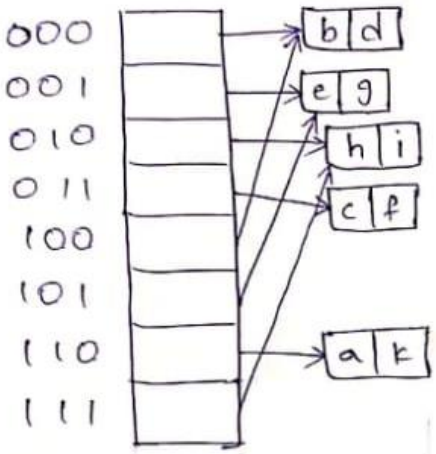
e) delete 24

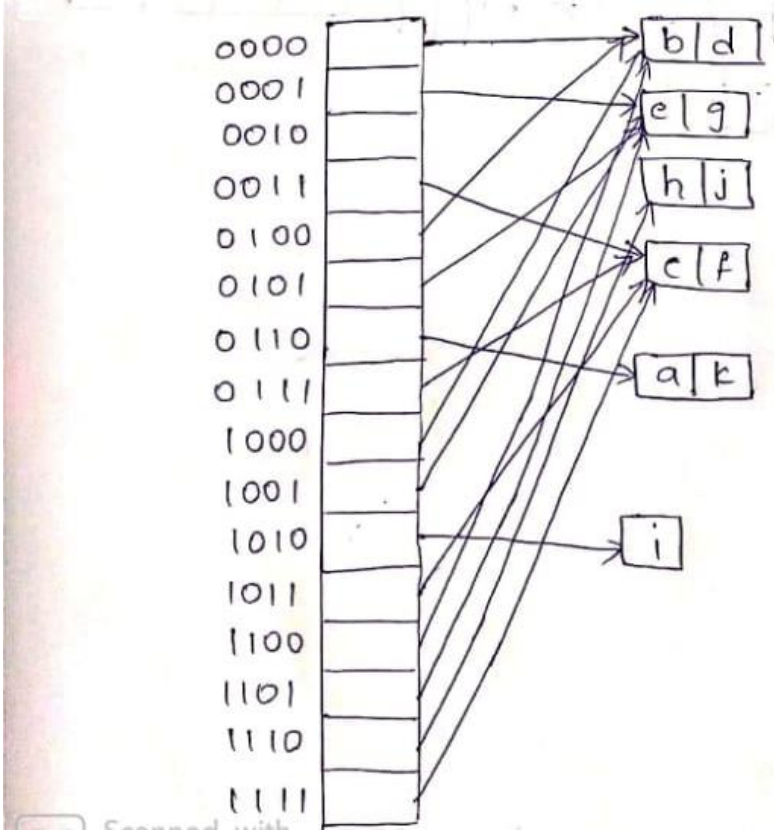


2) Suppose that we are using extendible hashing on a file that contains records with following search-key values: a [000110], b [111100], c [010111], d [010000], e [101001], f [010111], g [101001],h [011010], i [011010], j [011010], k [001110]. Consider an extendible hash structure where buckets can hold up to two records. Initially the structure is empty. Show the extendible hash structure after these records (in the order shown above) have been inserted.

2)


00
01
10
11
b d
e g
a h
c f


000
001
010
011
100
101
110
111
b d
e g
h i
c f
a k


0000
0001
0010
0011
0100
0101
0110
0111
1000
1001
1010
1011
1100
1101
1110
1111
b d
e g
h j
c f
a k
i