Branch : CSE & IT **Batch : Hinglish**

Database Management System

File Org & Indexing

DPP 03

[MCQ]

1. The order of a leaf node in a B^+ tree is the maximum number of (value, data record pointer) pairs it can hold. Given that the block size is 1K bytes, data record pointer is 8 bytes long, the value field is 10 bytes long and a block pointer is 6 bytes, then what is the order of the leaf node?

(a) 53 (b) 54
(c) 55 (d) 56

[NAT]

2. The order of a node in B^+ tree is defined as the number of pointers it can hold. What is the maximum number of keys that a B^+ tree of order 4 and height 4 can have ? ___________

(Assume that the height of a root node is 1)

[MCQ]

3. Given a block can hold either 3 records or 10 key pointers. A database contains P records, then how many blocks do we need to hold the data file and the dense index?

(a) $\frac{P}{30}$ (b) $\frac{P}{3}$
(c) $\frac{13P}{30}$ (d) $\frac{P}{10}$

[NAT]

4. The order of an internal node in B^+ tree index is the maximum number of children it can have. Assume that a child pointer takes 6 bytes, the search field value takes 34 bytes and the blocks size is 2048 bytes. The order of the internal node is ________.

[NAT]

5. Assume a disk with block size B = 1024 Bytes, A block pointer is P_B = 12 bytes long and a record pointer is P_R = 18 bytes long. A file has 1,00,000 patients records of size 100 bytes. Suppose the file is ordered by the key field PID and we want to construct a secondary (dense) index on non-key field DeptID (14 bytes), then minimum of how many blocks are required to store index file assuming an unspanned organisation?

(a) 3000 (b) 3100
(c) 3125 (d) None of the above

[NAT]

6. The order of a node in B tree is the maximum number of block pointers it can hold. Given that the block size is 2K bytes, data record pointer is 8 bytes long, the search key is 9 bytes long and a block pointer is 5 bytes long. The best possible order of B tree node is_____.

[NAT]

7. The order of a leaf node (P) in a B^+ tree is the maximum number of (value, data record pointer) pairs it can hold. Given that P=36, data record pointer is 8 bytes long, the search field is 6 bytes long and a block pointer is 8 bytes long. The permissible block size is_______.

[NAT]

8. (Assume that the level of root node is1)

The order of different nodes in B^+ tree/B tree are given as-

2 to P block pointers in root node.

$\left\lceil \frac{P}{2} \right\rceil$ to P block pointers in internal node.

$\left\lceil \frac{P}{2} \right\rceil$–1 to (P–1) keys in leaf node.

Let a and b be

The minimum number of keys in

B tree and B^+ tree node of order

P = 5 and level = 5. The value of (a + b) is______.

[NAT]

9. (Assume that the level of root node is1)

The order of different nodes in B^+ tree/B tree are given as-

2 to P block pointers in root node.

$\left\lceil \frac{P}{2} \right\rceil$ to P block pointer is internal node.

$\left\lceil \frac{P}{2} \right\rceil$–1 to (P–1) keys in leaf node.

Let a and b be the maximum number of keys in B tree and B^+ tree node of order P = 5 and level = 5. The value of (a +b) is________.

[NAT]

10. Consider the keys (1– 5000) are going to be interested into a B^+ tree. Assume, all the order are available before insertion. The orders P for B^+ tree node is defined as-

2 to P pointer for root

$\left\lceil \frac{P}{2} \right\rceil$ to P pointer for another node.

The maximum possible levels in a B^+ tree index for P = 9 is__________.

(Assume that level of the root node is 1)

[MCQ]

11. Consider the following statements:

S_1: In a B^+ tree, data pointers are stored only at the leaf nodes of the tree.

S_2: The leaf node has an entry for every value of the search field, along with the data pointer to the record.

Choose the correct statements.

(a) Only S_1 is true
(b) Only S_2 is true
(c) Both S_1 and S_2 are true
(d) Neither S_1 nor S_2 is true

[MSQ]

12. Which of the following is/are true reading B^+ tree?

(a) Records can be fetched in equal number of disk access.
(b) Height of the tree remains balanced and less as compared to B tree.
(c) Keys are used for indexing
(d) Faster search queries as the data is stored only on the leaf nodes.

[NAT]

13. Consider the keys (1– 5000) are going to be interested into a B^+ tree. Assume, all the order are available before insertion. The orders P for B^+ tree node is defined as-

2 to P pointer for root

$\left\lceil \frac{P}{2} \right\rceil$ to P pointer for another node.

The minimum possible levels in a B^+ tree index for P = 9 is__________.

(Assume that level of the root node is 1)

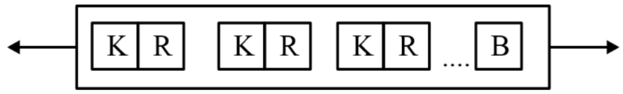
Answer Key

1. (d)
2. (255)
3. (c)
4. (52)
5. (c)
6. (93)
7. (512)
8. (269)
9. (5624)
10. (6)
11. (c)
12. (a, b, c, d)
13. (4)

Hints & Solutions

1. (d)

Disk block



Given data,

Disk block size = 1K byte = 2^{10} bytes = 1024 bytes

Block pointer (B) = 6 bytes

Key field (K) = 10 bytes

Record/ data pointer (R) = 8 bytes

Order of leaf node= P

$B + (P)(K + R) \leq D$

$6 + (P)(10 + 8) \leq 1024$

$18\,P \leq 1024 - 6$

$P = \left\lfloor \frac{1018}{18} \right\rfloor$

$\therefore$ P = 56

Maximum number of (value, data record pointer)

Pairs = 56

The order of the leaf node is 56.

2. (255)

A B^+ tree of order n and height h can have at most $n^h - 1$ keys. Therefore, maximum number of keys-

$= 4^4 - 1 = 255$.

3. (c)

For storing the records, numbers of blocks required = $\frac{P}{3}$ and for storing the keys in dense index number of blocks required = $\frac{P}{10}$.

So, total blocks required are $\frac{P}{3} + \frac{P}{10} = \frac{13\,P}{30}$

4. (52)

Size of child pointer = 6 bytes

Size of search field value = 34 bytes

Block size = 2048.

Order of internal node = P

($\therefore$ Number of blocks pointer in any node)

$(P - 1)34 + P \times 6 \leq 2048$

$34P + 6P \leq 2048 + 34$

$40P \leq 2082$

$P \leq \frac{2082}{40}$

$= \lfloor 52.05 \rfloor \simeq 52$

5. (c)

Blocking factor, bfr = $\lfloor 1024/100 \rfloor$

= 10 records per block

Number of blocks needs for file = $\lceil r/bfr \rceil$

$= \lceil 100000/10 \rceil = 10000$

Index records size R_i= (Non – Key DeptID + P_R))

= 14 + 18 = 32 bytes

Index blocking factors bfri = $\lfloor B/R_i \rfloor = \lfloor 1024/32 \rfloor$

= 32

Number of 1st level index entries r_1 = number of – records in the file = 100000 entries.

Number of first level index blocks $b_1 = \lceil r_1/bfri \rceil$

$= \lceil 100000/32 \rceil = 3125$ blocks

6. (93)

Order P: maximum blocks pointers per node.

Block size $\geq$ P × (Block size pointer) + (P –1) × (size of keys + size of record pointers)

Block size $\geq P \times 5 + (P - 1) \times (9 + 8)$

$2048 \geq 5P + 17P - 17$

$22P \leq 2065$

$P = \left\lfloor \frac{2065}{22} \right\rfloor = 93$

7. (512)

Order P: maximum number of (value, data record pointer) pairs

Block size $\geq$ P × (keys size + Record pointer size) + 1× (Block pointer size)

Block size $\geq P \times (6 + 8) + 1 \times (8)$

Block size $\geq 14 * 36 + 8$

Block Size = 512 bytes

8. (269)

Level	Minimum Number of Nodes	Minimum number of Blocks pointer	Minimum number of keys
1	1	2	1
2	2	$2 \times \left\lceil \frac{5}{2} \right\rceil = 6$	2 × 2
3	6	6 × 3 = 18	6 × 2
4	18	18 × 3 = 54	18 × 2
5	54	54 × 3 = 162	54 × 2

a = minimum number of keys in B tree $\rightarrow$ 161

For a B^+ tree, keys are present in last level only b = 108

$\therefore \quad a + b = 161 + 108 = 269$

9. (5624)

Level	Max. No. of nodes	Max. No. of Blocks pointer	Max. No. of keys
1	1	5	4
2	5	5 × 5	5 × 4
3	25	25 × 5	25 × 4
4	125	125 × 5	125 × 4
5	625	625 × 5	625 × 4

a = maximum number of keys in B tree → 3124

For a B^+ tree, keys are present in last level only b = 2500.

10. (6)

For maximum possible levels, minimum number of keys should be present in an index node.

Number of nodes in the last level = $\left\lfloor \frac{5000}{4} \right\rfloor = 1250$

[Minimum $\left(\left\lceil \frac{9}{2} \right\rceil - 1\right)$ keys for other node]

1 Node → root → $\lfloor 2/2 \rfloor$ = 1 node

2 Nodes → $\lfloor 10/5 \rfloor$ = 2 nodes

10 Nodes → $\lfloor 50/2 \rfloor$ = 10 nodes

50 Nodes → $\lfloor 250/5 \rfloor$ = 50 nodes

250 Nodes → $\lfloor 1250/5 \rfloor$ = 250 nodes

1250 Nodes

11. (c)

S_1(True): In a B^+ tree, data pointers are stored only at the leaf nodes of the tree.

S_2(True): the leaf nodes have an entry for every value of the search field, along with the data pointer to the record.

12. (a, b, c, d)

True: Records can be fetched in equal number of accesses

True: Height of the tree remains balanced and less as compared to B tree.

True: We can access the data stored in a B^+ tree sequentially as well.

True: Faster search queries as the data is stored only on the leaf node.

13. (4)

For minimum possible levels, maximum number of keys should be present in index node.

$= \left\lceil \frac{5000}{8} \right\rceil = 625$

$= \left\lceil \frac{625}{9} \right\rceil = 70$

$= \left\lceil \frac{70}{9} \right\rceil = 8$ Minimum number of level = 4

$= \left\lceil \frac{8}{9} \right\rceil = 1$ node