International Islamic University Chittagong

Department of Computer Science and Engineering

B. Sc. in CSE, Final Assessment Test, Spring 2021

Course Code: CSE 2423 Course Title: Database Management Systems

Section: 4AM

Total marks: ***30***

Time: ***4*** *hours* ***30*** *minutes for* ***exam*** *+* ***30*** *minutes for* ***submission***

[Answer all the questions;
Precisely follow the guideline for preparing and submitting the answer script;
Figures in the right hand margin indicate full marks.]

1. An agency called Instant Cover supplies part-time/temporary staff to hotels in Scotland. Figure 1 lists the time spent by agency staff working at various hotels. The national insurance number (NIN) is unique for every member of staff. Use Figure 1 to answer questions (a) and (b). 6

NIN	ContractNo	Hours	eName	hNo	hLoc
1135	C1024	16	Smith J.	H25	East Killbride
1057	C1024	24	Hocine D.	H25	East Killbride
1068	C1025	28	White T.	H4	Glasgow
1135	C1025	15	Smith J.	H4	Glasgow

Figure 1

a. This table is susceptible to update anomalies. Provide examples of insertion, deletion and update anomalies.

b. Normalize this table to third normal form. State any assumptions.

2. Construct a B+ tree for the following set of values (2, 3, 5, 7, 11, 17, 19, 23, 29, 31). Assume that the tree is initially empty and values are inserted in ascending order. 6

a) Construct B+-trees for the cases where the number m of pointers that will fit a node is as follows: a. Four b. Seven

b) Shows the form of the B+-tree after each operation of the sequence : Insert 9; Insert 10; Insert 8; Insert 6; Insert 1; Insert 4 for the case m=4.

3. Indicate whether the following schedules can produce anomalies; the symbols ci and ai indicate the result (commit or abort) of the transaction. 6

(a) r1(x); w1(x); r2(x); w2(y); w3(x); r3(y); w3(y); a1; c2; c3
(b) r1(x); w1(x); r2(y); w2(y); w3(x); r3(y); w3(y); a1; c2; c3
(c) r1(x); r2(x); r2(y); w2(y); r1(z); w3(x); r3(y); w3(y); a1; c2; c3
(d) r1(x); r2(x); w3(x); r3(y); w3(y); w2(x); r1(y); a1; c2; c3

4. For information on multivalued dependencies and 4NF, review the slides. 6

Suppose we have the following Courses table with columns CourseID, Instructor, Book that stores the courses, the instructor teaching the course, and the recommended books for the course. The book(s) recommended for a course does not depend on the teacher teaching the course, just on the course. Here is an example of instantiation for this table:

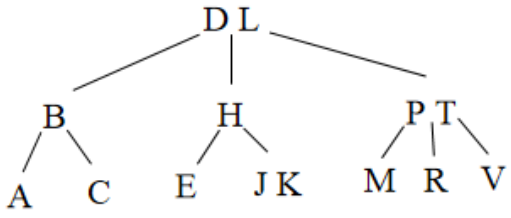
Courses:

CourseID	Instructor	Book
IT360	Crainiceanu	Kroenke
IT360	Crainiceanu	Welling
IT360	DeLooze	Kroenke
IT360	DeLooze	Welling
SI440	Crainiceanu	Kroenke
SI440	Crainiceanu	Ramakrishnan
SI440	Crainiceanu	Stonebraker

a) Give an example of a multivalued dependency in the Courses table.

b) Is the Courses table in 4NF? If answer to yes, say why. If not, decompose the table such that the resulting tables are in 4th normal form. For each of the resulting tables, give the table name, column names, primary keys, and foreign keys.

5. Show the B-tree the results when deleting A, then deleting V and then deleting P from the following B-tree with a minimum branching factor of t=2. 6



Ans to the question no. 1(a)

Insertion anomaly: An insertion anomaly is the inability to add data to the database due to absence of other data.

Example: If a new Hotel hNO is inserted into the table, it is necessary to add the NIN details even though hotel has not taken the help of agency to hire employes.

Deletion anomaly: A deletion anomaly occurs when delete a record that may contain attributes that shouldn't be deleted.

Example: If a hotel hNo is closed and is deleted from table, the corresponding details of NIN will also get deleted.

Update anomaly: An update anamoly is a data inconsistency that results from data redundancy and a partial update.

Example: If hotel location is changed, it needs to be updated in all the rows, failing which can cause inconsistancy in data.

Ans to the question no. 1(b)

Given table already in 1NF because it has no NULL value and all columns has atomic entities.

For 2NF:

'Table-1'

NIN	Contract No	Hours
1135	C1024	16
1057	C1024	24
1068	C1025	28
1135	C1025	15

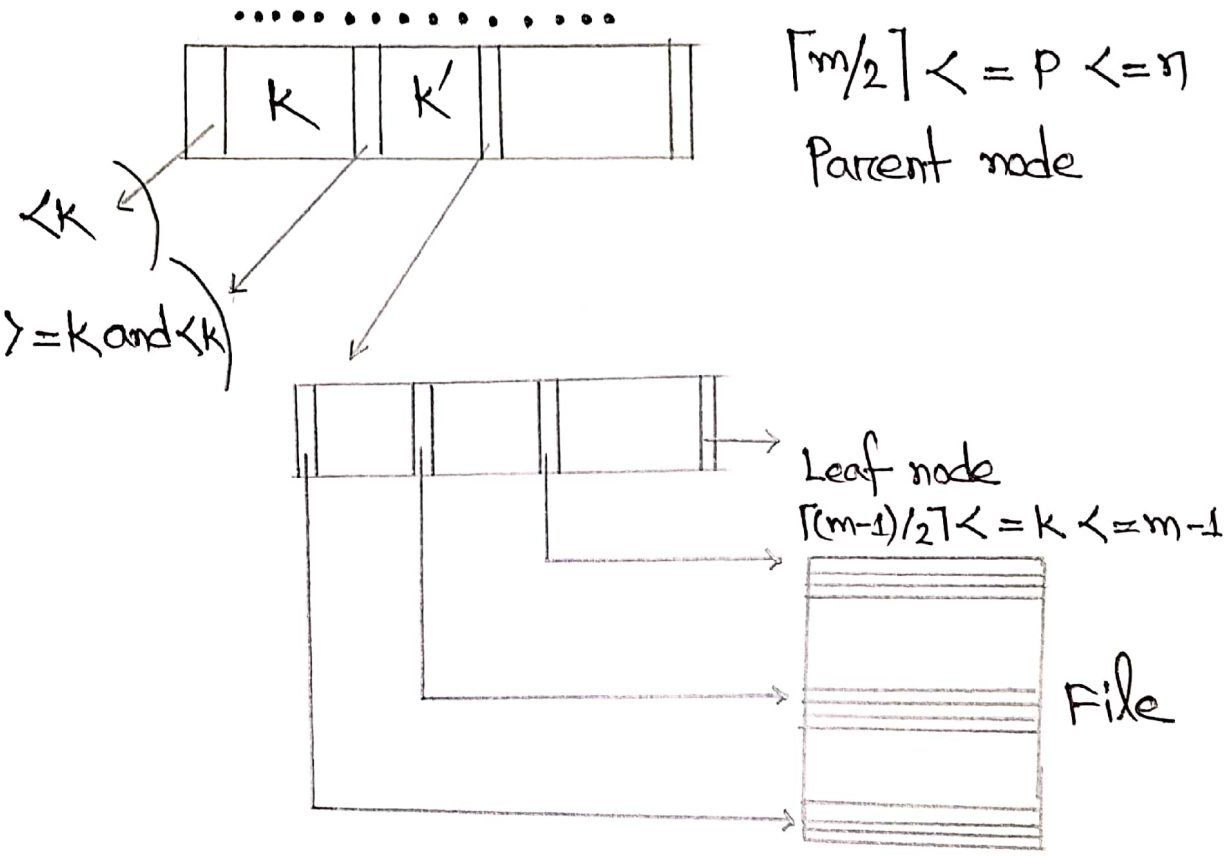
'Table-2'

Contract No.	hNo	hLoc
C1024	H25	East Kilbride
C1025	H4	Glasgow

'Table-3'

NIN	eName
1135	Smith J.
1057	Hocine D.
1068	white T.

Ans to the question no. 2(a)



B+-tree:

Not a root or a leaf node: $\lceil m/2 \rceil <= p <= m$, where p is pointer.

Leaf node:

$\lceil (m-1)/2 \rceil <= k <= m-1$, where k is a key

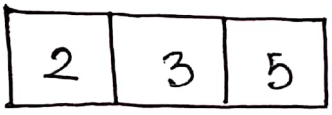
Root: $p >= 2$ if it is not a leaf

$0 <= k <= (m-1)$ if its a leaf

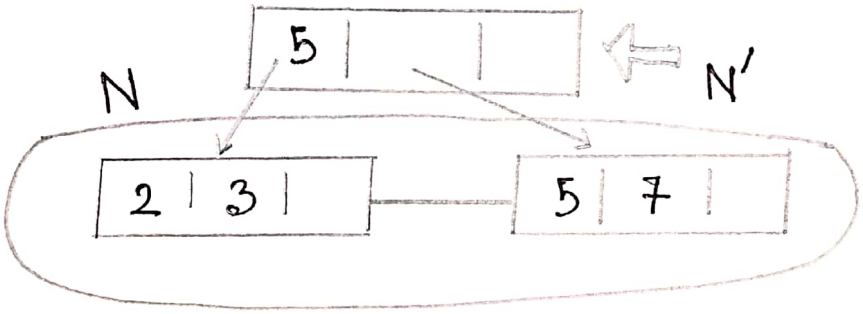
a) $m = 4$ Internal node : $2 <= P <= 4$ Root: $P >= 2$ if it is not a leaf

Leaf node: $2 <= k <= 3$ $0 <= k <= 3$ if it is a leaf

B+ tree: Insert 2,3,5



B+ tree: Insert 7 (leaf split)

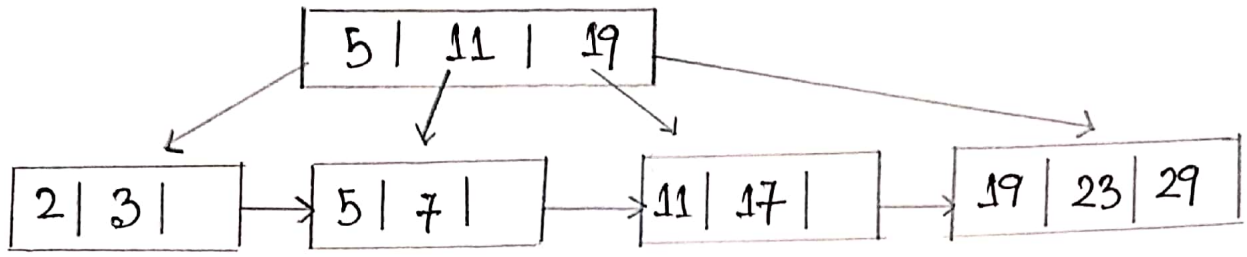


split a leaf node:

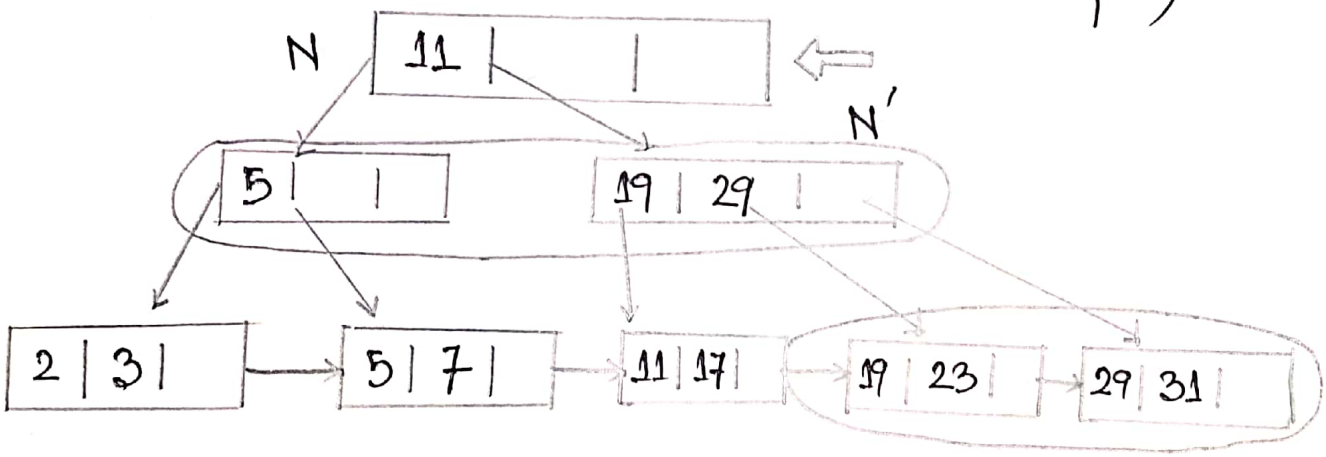
Let $k_1 \ldots . k_m$ be the set of keys in the ascending order (i.e. 2,3,5,7)

- Node N : $k_1 \ldots . k_{\lceil m/2 \rceil - 1}, k_{\lceil m/2 \rceil}$ (i.e 2,3)
- Node N': $k_{\lceil m/2 \rceil + 1} \ldots . k_m$ (i.e. 5,7)
- Insert $k_{\lceil m/2 \rceil + 1}$ into the parent node (i.e. 5)

B+ tree: Insert 11, 17, 19, 23, 29



B+ tree: Insert 31 (leaf node split & non leaf node split)



Split on internal node N:

Let $k_1 \ldots\ldots k_3$ be the set of keys in the ascending order (i.e. 5, 11, 19, 29)

- Node N: $k_1 \ldots\ldots k_{\lceil m/2 \rceil - 1}$ (i.e. 5)
- Node N': $k_{\lceil m/2 \rceil + 1} \ldots\ldots k_m$ (i.e. 19, 29)
- Insert $k_{\lceil m/2 \rceil}$ into the parent node of N (i.e. 11)

Note that since values are inserted in ascending order, leaves have the minimum number of keys except the last leaf.

b)

Seven,

$m = 7$ Intermediate node: $4 <= p <= 7$

Leaf Node: $3 <= k <= 6$

Root: $p >= 2$, if it is not a leaf.

$0 <= k <= 6$ it is a leaf.

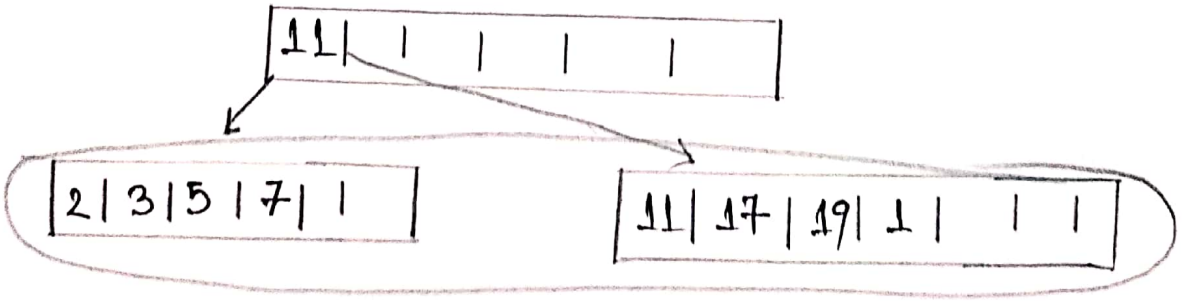
B+ tree:

Insert 2, 3, 5, 7, 11, 17

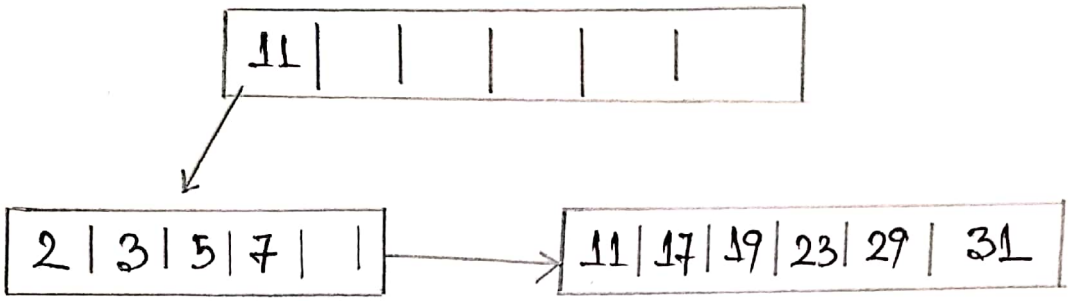
2	3	5	7	11	17

B+ tree:

Insert 19 (leaf split)

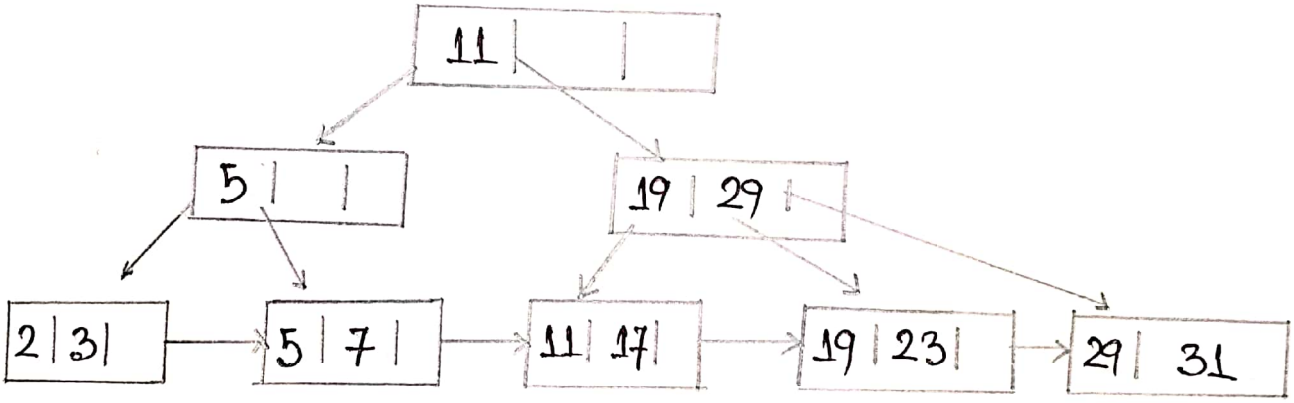


B+ tree : Insert 23, 29, 31

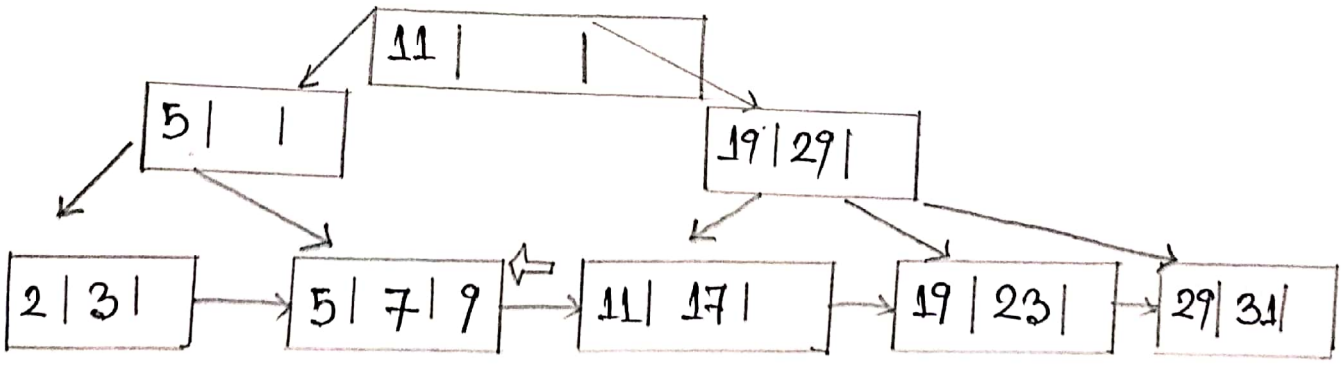


Ans to the question no. 2(b)

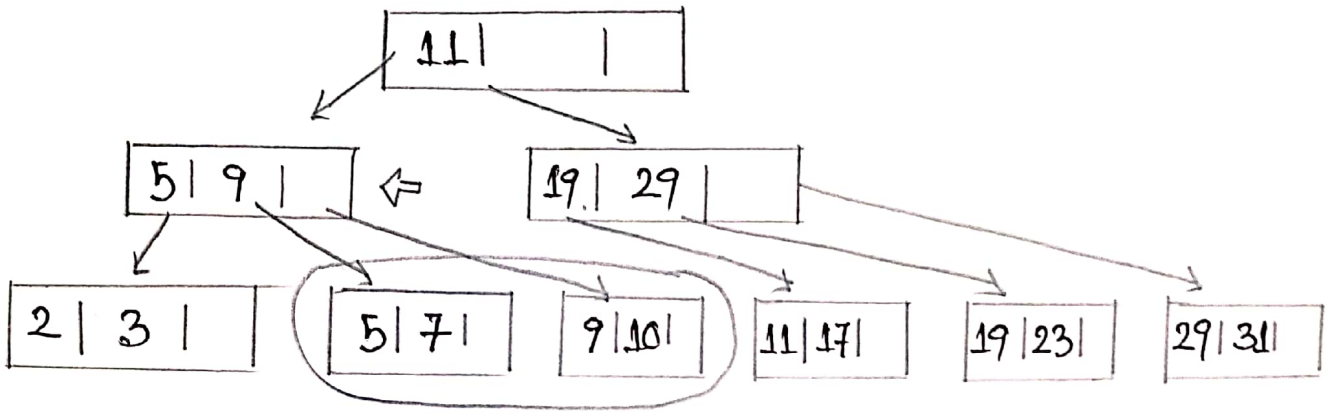
B+ tree



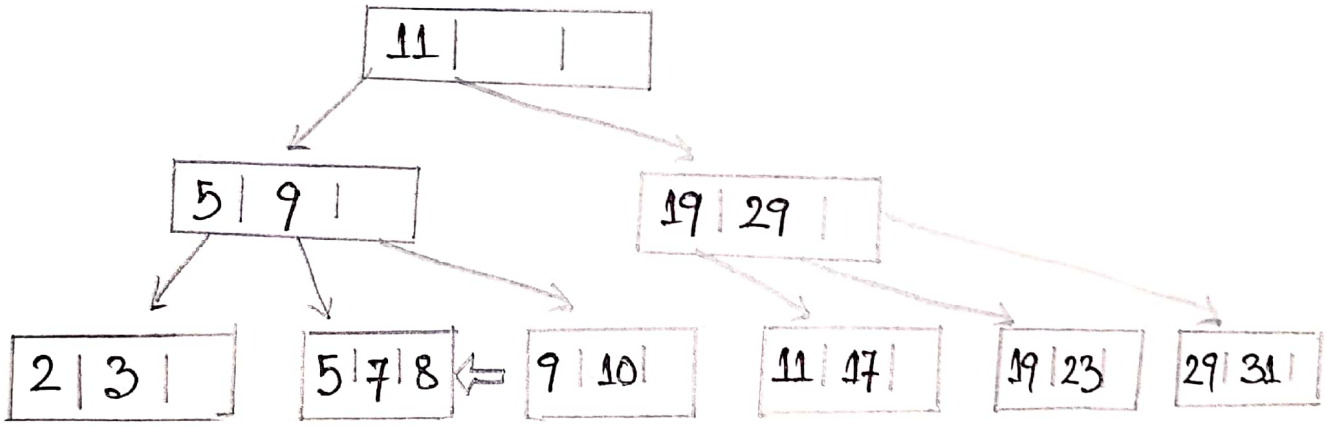
B+ tree : Insert 9



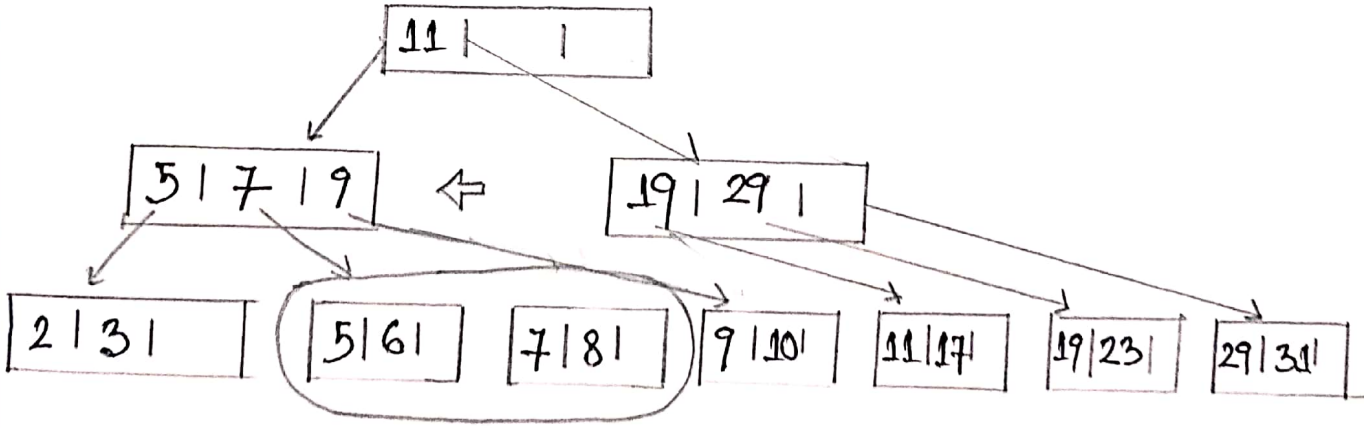
B+ tree : Insert 10 (leaf split)



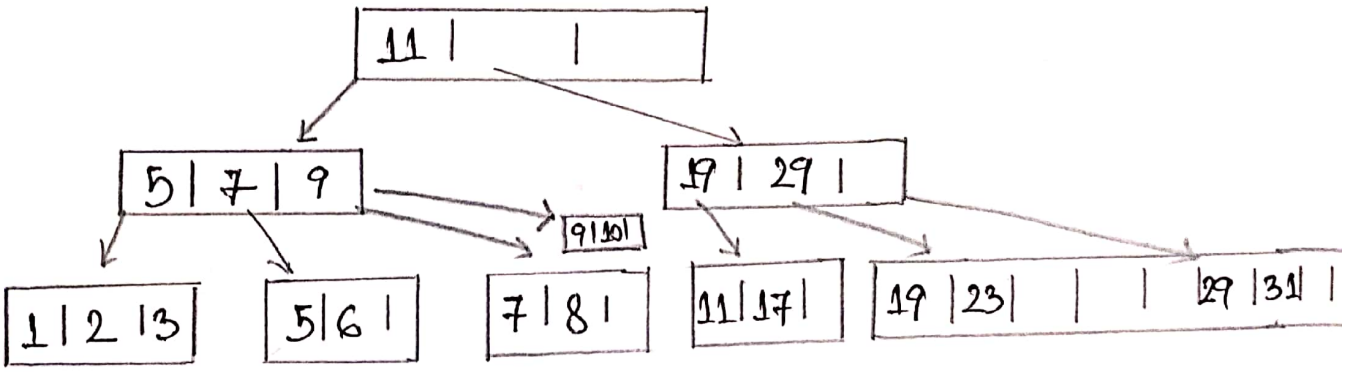
B+ tree : Insert 8



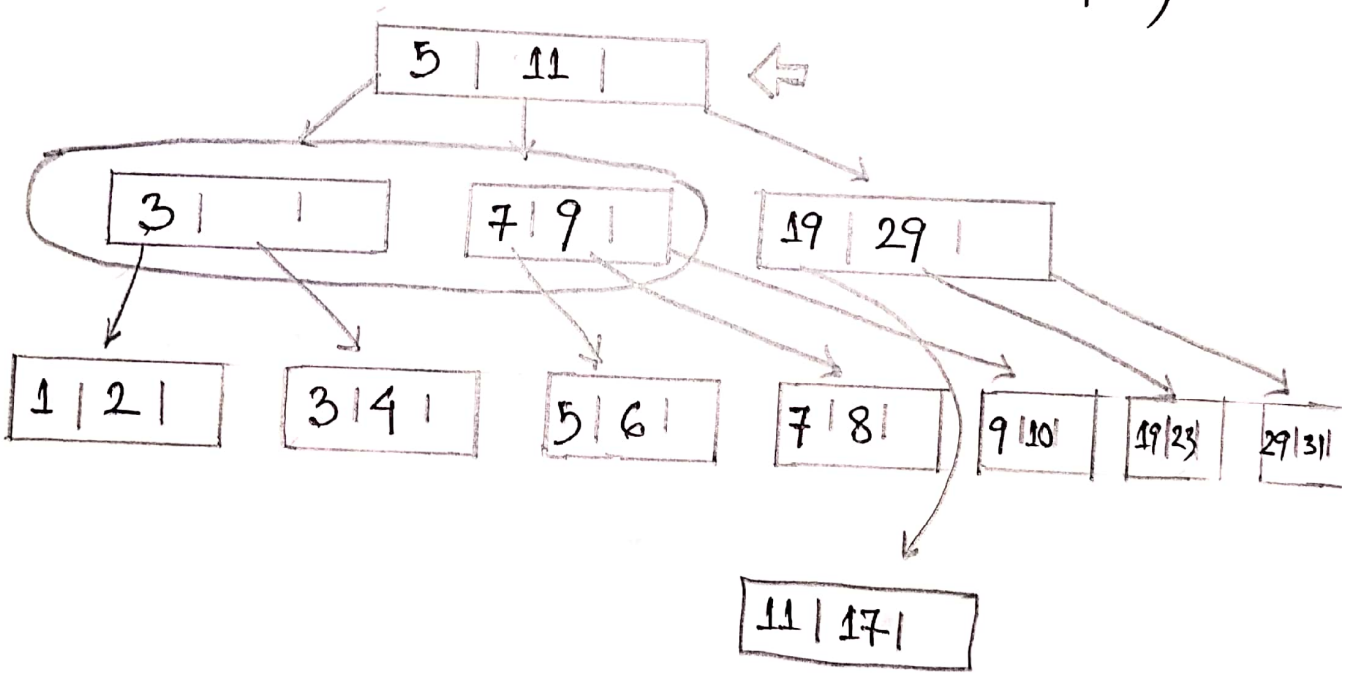
B+ tree : Insert 6 (leaf split)



B+ tree: Insert



B+ tree: Insert 4 (leaf node split + non leaf node split)



Ans to the question no-03

a) Given schedule is:

$r_1(x)$; $w_1(x)$; $r_2(x)$; $w_2(y)$; $w_3(x)$; $r_3(y)$; $w_3(y)$; a1; c2; c3

the schedule produce write-write anomalies and the anomalies underlined below:-

$r_1(x)$; $\underline{w_1(x)}$; $r_2(x)$; $w_2(y)$; $\underline{w_3(x)}$; $r_3(y)$; $w_3(y)$; $\underline{a_1}$; c2; c3

b) Given schedule is:

$r_1(x)$; $w_1(x)$; $r_2(y)$; $w_2(y)$; $w_3(x)$; $r_3(y)$; $w_3(y)$; a1; c2; c3

The schedule produce write-write anomalies and the anomalies underlined below:-

$r_1(x)$; $\underline{w_1(x)}$; $r_2(y)$; $w_2(y)$; $\underline{w_3(x)}$; $r_3(y)$; $w_3(y)$; $\underline{a_1}$; c2; c3

c) Given schedule is:

$r_1(x)$; $r_2(x)$; $r_2(y)$; $w_2(y)$; $r_1(z)$; $w_3(x)$; $r_3(y)$; $w_3(y)$; a_1; c_2; c_3;

This schedule has no anomalies.

d) Given schedule is :-

$r_1(x)$; $r_2(x)$; $w_3(x)$; $r_3(y)$; $w_3(y)$; $w_2(x)$; $r_1(y)$; $a1$; $c2$; $c3$

This schedule has no anomalies.

Ans to the question no. 4(a)

Multivalued dependency occured when two attributes in a table are independent of each other but, both depend on a third attribute. In this given table Instructor and Book are dependent on CourseID and independent each other. So the dependency is shown below:

CourseID →→ Instructor

CourseID →→ Book

Ans to the question no. 4(b)

Given Courses table is in 3NF. but Instructor and Book are two independent entity. Hence there is no relation between Instructor and Book So, if we make the 'Course' table into 4NF, we can decompose it into two tables.

Course_Instructor:

CourseID	Instructor
IT360	Crainiceanu
IT360	DeLooze
SI440	Crainiceanu

"CourseID" is the primary key.

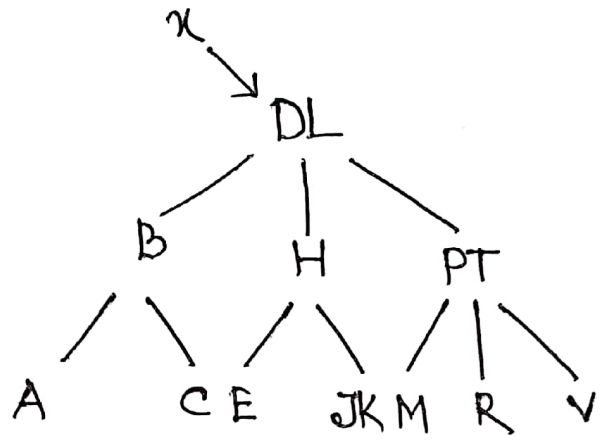
Course Book

CourseID	Book
IT360	Kroenke
IT360	Welling
SI440	Kroenke
SI440	Ramakrishnan
SI440	Stonebraker

"CourseID" is the primary key.

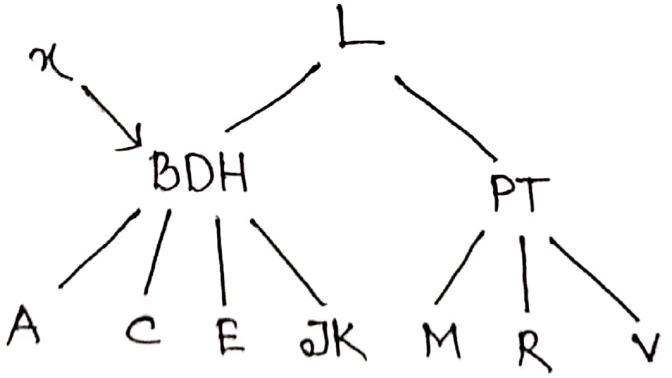
Ans to the question no. 05

The B-tree results when deleting A, then deleting V and deleting P from given B-tree with a minimum branching factor of $t=2$ is shown below—



deleting 'A'

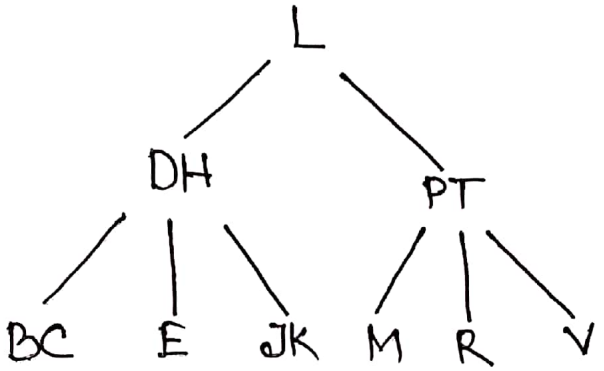
Applying Case (3b)



deleting 'A' →

Applying (3b) then

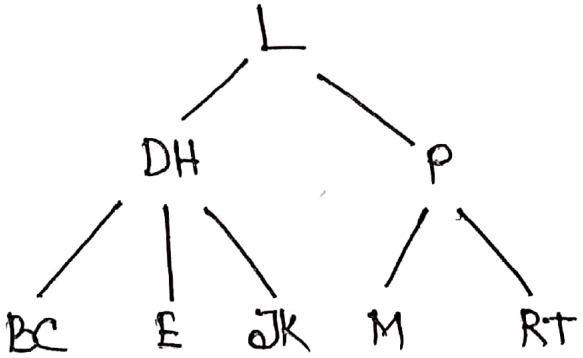
Case 1



deleting 'V' go to →

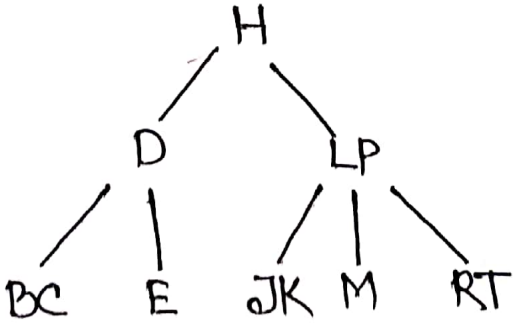
pt then Applying Case(3b)

and case 1



deleting 'P' →

Applying 3a



deleting 'P'

Case 2b