



VIT[®]
UNIVERSITY
(Estd. u/s 3 of UGC Act 1956)

School of Computer Science and Engineering Digital Assignment – I Winter – SEMESTER 2022 - 23

Course Code: MCSE506L

Course-Title: – Database Systems

Programme Name: M.Tech

Branch: CSE with Specialization in AI & ML

Class Number (s): VL2022230506179

SLOT: B2+TB2

Faculty: Dr. KARTHIK G M - SCOPE

Name: Nidhi Singh

Reg. No: 22MAI0015

-
1. A. Suppose that we decompose the schema $R = (A, B, C, D, E)$ into (A, B, C) , (A, D, E) . Show that this decomposition is a lossless-join decomposition if the following set F of functional dependencies holds: $A \rightarrow BC$, $CD \rightarrow E$, $B \rightarrow D$, $E \rightarrow A$

Answer :-

Q-1.A

$$R = (A, B, C, D, E)$$

decomposed into (A, B, C) , (A, D, E)

FD:

$$A \rightarrow BC$$

$$CD \rightarrow E$$

$$B \rightarrow D,$$

$$E \rightarrow A$$

To check for lossless join decomposition using FD set, following condition must hold :-

Given decomposition is lossless join

iff

$$\textcircled{1} \quad R_1 \cup R_2 = R$$

$$\textcircled{2} \quad \begin{aligned} R_1 \cap R_2 &\neq \emptyset \\ R_1 \cap R_2 &\rightarrow R_1 \\ \text{OR} \\ R_1 \cap R_2 &\rightarrow R_2 \end{aligned} \quad \left. \begin{array}{l} R_1 \cap R_2 \text{ is SK of } R, \text{ or } R_2 \\ \text{OR} \\ R_1 \cap R_2 \text{ is PK of } R, \text{ or } R_2 \end{array} \right\}$$

$$R_1 (A, B, C)$$

$$R_2 (A, D, E)$$

$$\begin{aligned} \textcircled{1} \quad R_1 \cup R_2 &= \{A, B, C\} \cup \{A, D, E\} \\ &= \{A, B, C, D, E\} \end{aligned}$$

$$R_1 \cup R_2 = R$$

\hookrightarrow satisfied

$$\textcircled{2} \quad R_1 \cap R_2 \neq \emptyset$$

$$\{A, B, C\} \cap \{A, D, E\} \neq \emptyset$$

$$\{A\} \neq \emptyset$$

satisfied

$$R_1 \cap R_2 \rightarrow \{A, B, C\} \cap \{A, D, E\}$$

$$R_1 \cap R_2 \rightarrow A$$

Now check A is Super Key of R_1 or R_2

$$A^+ = ABC$$

A is SuperKey of $R_1(A, B, C)$

Hence given decomposition $R_1(A, B, C)$ & $R_2(A, D, E)$

is lossless join decomposition.

B. List all functional dependencies satisfied by the relation.

A	B	C
a1	b1	c1
a1	b1	c2
a2	b1	c1
a2	b1	c3

1. B

The nontrivial functional dependencies are :-

$A \rightarrow B$ and (\rightarrow) (one to one relationship)

$C \rightarrow B$ (one to many)

and a dependencies they logically imply :-

$AC \rightarrow B$

There are 19 trivial functional dependencies of the form $\alpha \rightarrow \beta$, where $\beta \subseteq \alpha$.

C does not functionally determine A because the first and third tuples have the same C but different A values.

The same tuple also show B does not functionally determine A.

Likewise, A does not functionally determine C because the first two tuples have the same A Value and different C Values.

The same tuples also show B does not functionally determine C.

2. Suppose a relational schema R ($A B C D E F G H I$) and set of functional dependencies F : { $AB \diamond C$, $AD \diamond GH$, $BD \diamond EF$, $A \diamond I$, $H \diamond J$ }. Check out that relation is in 3NF or not? If not decompose it in 3NF.

(2)

Rule for 3NF

A relation is said to be 3NF, if it holds at least one of the following for every non-trivial functional dependency $\alpha \rightarrow \beta$:

- $\Rightarrow \alpha$ is Super Key
- $\Rightarrow \beta$ is prime attribute.

Given relation:-

$$F: AB \rightarrow C$$

$$AD \rightarrow GH$$

$$BD \rightarrow EF$$

$$A \rightarrow I$$

$$H \rightarrow J$$

$$R(A B C D E F G H I)$$

Candidate Key :-

$$(ABD)^+ = ABCDEFIGHI$$

$$H \rightarrow I$$

$$H \rightarrow J$$

Candidate Key :-

$$(ABD)^+ = ABCDEFGHI$$

ABD is the Candidate key, because closure of ABD has all the attributes of R.

Prime Attribute $\Rightarrow A, B, D$

Non Prime Attribute $\Rightarrow C, E, F, G, H, I$

$$AB \rightarrow C$$

\hookrightarrow not superkey, & non C is prime attribute

Partial dependency

not in 3NF, not in 2NF

$$AD \rightarrow GH$$

AD is not superkey, non GH is prime attribute

AD is partial dependency

not in 3NF, not in 2NF

$BD \rightarrow EF$

BD is not super key

EF is not prime attribute

Partial dependency

not in 2NF & not in 3NF

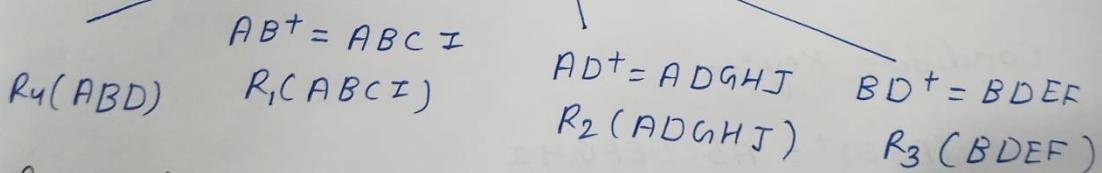
$A \rightarrow I$ not in 3NF & not in 2NF

$H \rightarrow J$ not in 3NF & But it is in 2NF
 $\hookrightarrow 2NF$

So Relation $R(A, B, C, D, E, F, G, H, I, J)$ is not in 3NF

as it is not following the rules of 3NF.

Therefore, $R(ABCDEFGHIJ)$ need to be decomposed:



$R_4(ABD) \rightarrow$

ABD is super key

Relation R_4 is in 3NF

\hookrightarrow is in 3NF

$R_1(ABCIZ)$

$AB \rightarrow C$

$AB^+ = ABCI$

AB is superkey
is in 3NF

$A \rightarrow I$

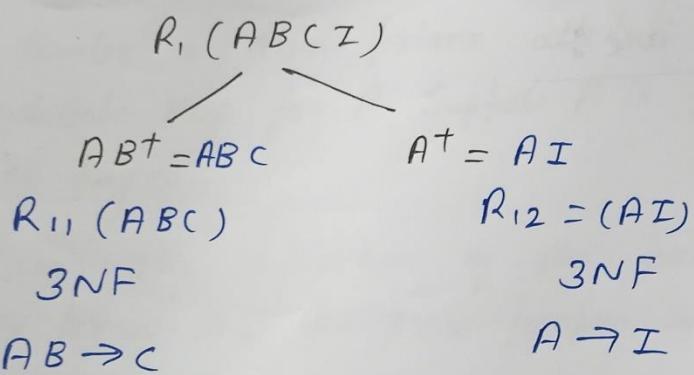
A is not superkey and I is not prime attribute
Partial dependency

not in 3NF & 2NF.

Need to decompose the Relation

$R_1(ABCIZ)$

$\swarrow \qquad \searrow$



$R_{11} (ABC)$ $\nearrow 3NF$
 $R_{12} (AI)$

$R_2 (ADGH\bar{IJ})$
 $AD^+ = ADGHJ$
 $AD \rightarrow GH$
 \hookrightarrow Super Key
 is in 3NF

$H \rightarrow J$

note H is not Super Key
 J is not prime attribute
 Not in 3NF but in 2NF

$R_2 (ADGH\bar{IJ})$
 $\swarrow \quad \searrow$
 $R_{21} (ADGH)$ $R_{22} (H\bar{J})$
 $AD^+ = ADGH$ $H^+ = HJ$
 $AD \rightarrow GH$ $H \rightarrow J$
 3NF 3NF

$R_3 (BDEF)$
 $BD^+ = BDEF$
 $BD \rightarrow EF$
 BD is Super Key
 Relation R_3 is in 3NF.

3NF Relations are :-

$R_{11}(ABC)$

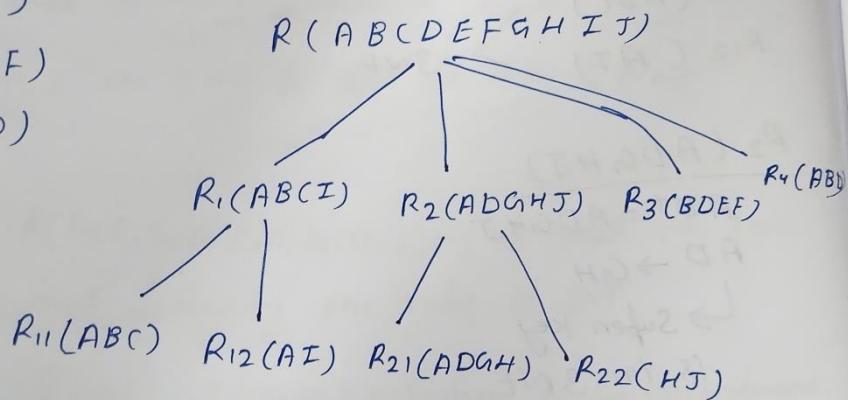
$R_{12}(AI)$

$R_{21}(ADGH)$

$R_{22}(HJ)$

$R_3(BDEF)$

$R_4(ABD)$



Now R_{11} , R_{12} , R_{21} , R_{22} , R_3 and R_4 are in 3NF.

3. A functional dependency $\alpha \rightarrow \beta$ is called a partial dependency if there is a proper subset γ of α such that $\gamma \rightarrow \beta$. We say that β is partially dependent on α . A relation schema R is in second normal form (2NF) if each attribute A in R meets one of the following criteria:

- It appears in a candidate key.
- It is not partially dependent on a candidate key.

Show that every 3NF schema is in 2NF.

Ans-3

A relational Schema R is said to be in 3NF if there is no non-prime attribute A in R for which A is transitively dependent on a key for R .

2NF

A Relational Schema R is in 2NF if no non-prime attribute A is partially dependent on any Candidate Key for R .

To prove that every 3NF schema is in 2NF, it suffices to show that if a non-prime attribute A is partially dependent on a candidate key α , then A is also transitively dependent on the key α .

Let A be a non-prime attribute in R . Let α be a candidate key for R . Suppose A is partially dependent on the key α .

\Rightarrow From the definition of the partial dependency, we know that for some proper subset β of α ,

$$\beta \rightarrow A$$

\Rightarrow Since $\beta \subset \alpha$, $\alpha \rightarrow \beta$. Also, $\beta \rightarrow \alpha$ does not hold.
Since α is a candidate key.

\Rightarrow Finally, Since A is non-prime, it cannot be in either β or α .

Thus we conclude that $\alpha \rightarrow A$ is a transitive dependency. Hence we have proved that every 3NF Schema is also in 2NF.

4. Let relations $r_1(A, B, C)$ and $r_2(C, D, E)$ have the following properties:
 r_1 has 20,000 tuples, r_2 has 45,000 tuples, 25 tuples of r_1 fit on one block, and 30 tuples of r_2 fit on one block. Estimate the number of block transfers and seeks required using each of the following join strategies for $r_1 \bowtie r_2$:
- a. Nested-loop join.

Ans - 4

r_1 needs 800 blocks, and r_2 needs 1500 blocks.
Let us assume M pages of memory. If $M > 800$, the join can easily be done in $1500 + 800$ disk accesses, using even plain nested loop join. So we consider only the case where $M \leq 800$ pages.

(a) Nested-loop join

Using r_1 as the outer relation we need
 $20000 * 1500 + 800 = 30,000,800$ disk accesses;
if r_2 is outer relation we need $45000 * 800 + 1500 = 36,001,500$ disk accesses.

- b. Block nested-loop join.

- c. Merge join.

(b) Block nested-loop join

If r_1 is the outer relation, we need $\lceil \frac{800}{M-1} \rceil * 1500 + 800$ disk accesses,
If r_2 is the outer relation we need
 $\lceil \frac{1500}{M-1} \rceil * 800 + 1500$ disk accesses.

(c) Merge-join

Assuming that r_1 and r_2 are not initially sorted on the join key, the total sorting cost inclusive of the output is B_S

$$B_S = 1500(2\lceil \log_{M-1}(1500/M) \rceil + 2) + 800(2\lceil \log_{M-1}(800/M) \rceil + 2)$$

disk accesses.

Assuming all tuples with the same value for the join attributes fit in memory, the total cost is $B_S + 1500 + 800$ disk accesses.

d. Hash join

e. Semi Join

(d) Hash-Join

We assume no overflow occurs. Since r_1 is smaller, we use it as the build relation and r_2 as the probe relation. If $M > 800/M$, i.e.

no need for recursive partitioning, then the cost is $3(1500 + 800) = 6900$ disk accesses,

else the cost is $2(1500 + 800) \lceil \log_{M-1}(800) - 1 \rceil + 1500 + 800$ disk accesses.

(e) Semi Join

Given that r_1 has 20000 tuples and 25 tuples fit on one block, we have:

f. Bloom Join

\Rightarrow Block transfer:

20000 tuple of r_1 / 25 tuples per block = 800 block
for r_1

(f) Bloom Join

Bloom join utilizes Bloom filters to pre-filter the tuples before joining. Let's assume a selectivity of 50% for the Bloom filters.

For r_1

$$\begin{aligned} \text{Number of bits in Bloom filter} &= 20000 \text{ tuples of } r_1 * 0.5 \\ &\quad (\text{Selectivity}) \\ &= 10000 \text{ bits} \end{aligned}$$

$$\begin{aligned} \text{Number of Bloom filter blocks} &= 10000 \text{ bits} / (25 \text{ tuples per} \\ &\quad \text{block} * 8 \text{ bits per byte}) \\ &= 50 \text{ blocks} \end{aligned}$$

$$= 10000 \text{ bits}$$

Number of Bloom filter blocks = $10000 \text{ bits} / (25 \text{ tuples per block} * 8 \text{ bits per byte})$
 $= 50 \text{ blocks}$

For τ_2

Number of bits in Bloom filter = $45000 \text{ tuples of } \tau_2 * 0.5$
 $= 22500 \text{ bits}$

Number of Bloom filter blocks = $22500 \text{ bits} / (30 \text{ tuples per block} * 8 \text{ bits per byte})$
 $= 94 \text{ blocks}$

Block transfers = 50 blocks for τ_1 + 94 blocks for τ_2

5. Check whether the given schedule S is conflict serializable or not. If yes, then determine all the possible serialized schedules.

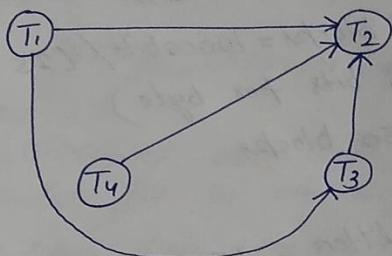
T1	T2	T3	T4
READ A WRITE B	READ A WRITE A WRITE B	READ A READ B	READ A

Answer :-

Hm8-5

T_1	T_2	T_3	T_4
Read A	Read A	Read A	Read A
Write B	Write A, Write B	Read B	

Precedence graph



Schedule S is Conflict Serializable Schedule

iff

"Precedence graph of Schedule (S) must be acyclic"

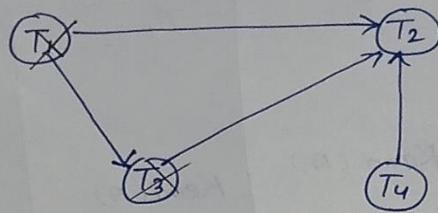
ie

Conflict equal Serial Schedules are topological
order of Schedule (S) Precedence graph.

There is no cycle in the graph that means the
Schedule is Conflict Serializable Schedule.

Possible Serialized Schedules

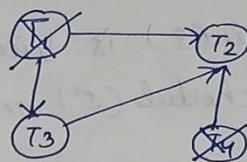
Topological order



$T_1 : T_3 : T_4 : T_2$

$T_1 : T_4 : T_3 : T_2$

$T_4 : T_1 : T_3 : T_2$



Serialized Schedule

①

T_1	T_2	T_3	T_4
Read(A) Write(B)	Write(A) Write(B)	Read(A) Read(B)	Read(A)

②

T_1	T_2	T_3	T_4
Read(A) Write(B)	Write(A) Write(B)	Read(A) Read(B)	Read(A)

③

T_1	T_2	T_3	T_4
Read(A) Read(A) Write(B)	Read(A) Write(B)	Read(A) Read(B)	Read(A)

6. Check whether the given schedule S is view serializable or not?

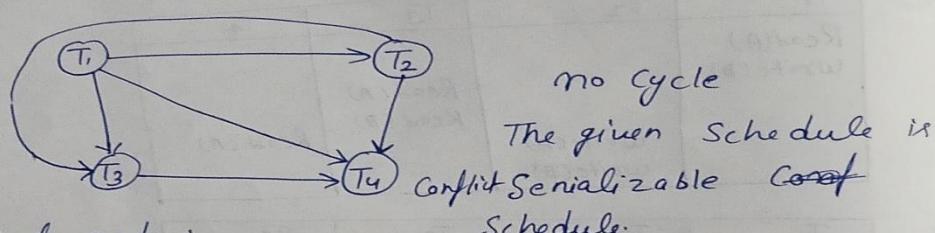
T1	T2	T3	T4
READ A WRITE B	READ A WRITE B	READ A WRITE B	READ A WRITE B

HM8-6

T ₁	T ₂	T ₃	T ₄
Read(A) write(B)	Read(A) write(B)	Read(A) write(B)	Read(A) write(B)

If Schedule (S) is View Serializable iff some Serial Schedule (S') must be view causal to Schedule (S).

If Schedule (S) Conflict Serializable then Schedule (S) is also view Serializable.



Topological order:

T₁ : T₂ : T₃ : T₄

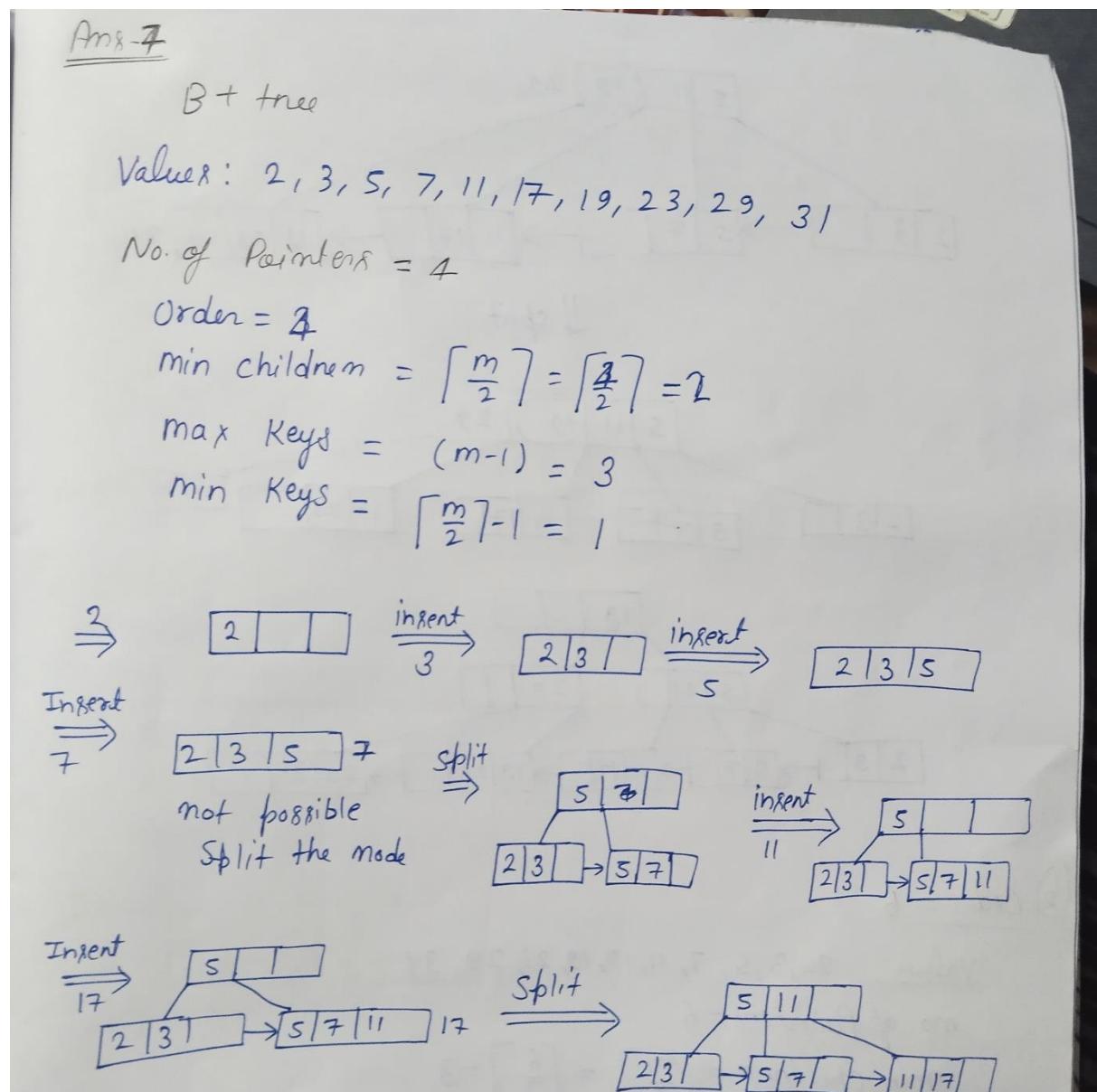
Schedule (S) is Conflict Serializable Schedule then the given Schedule is View Serializable Schedule.

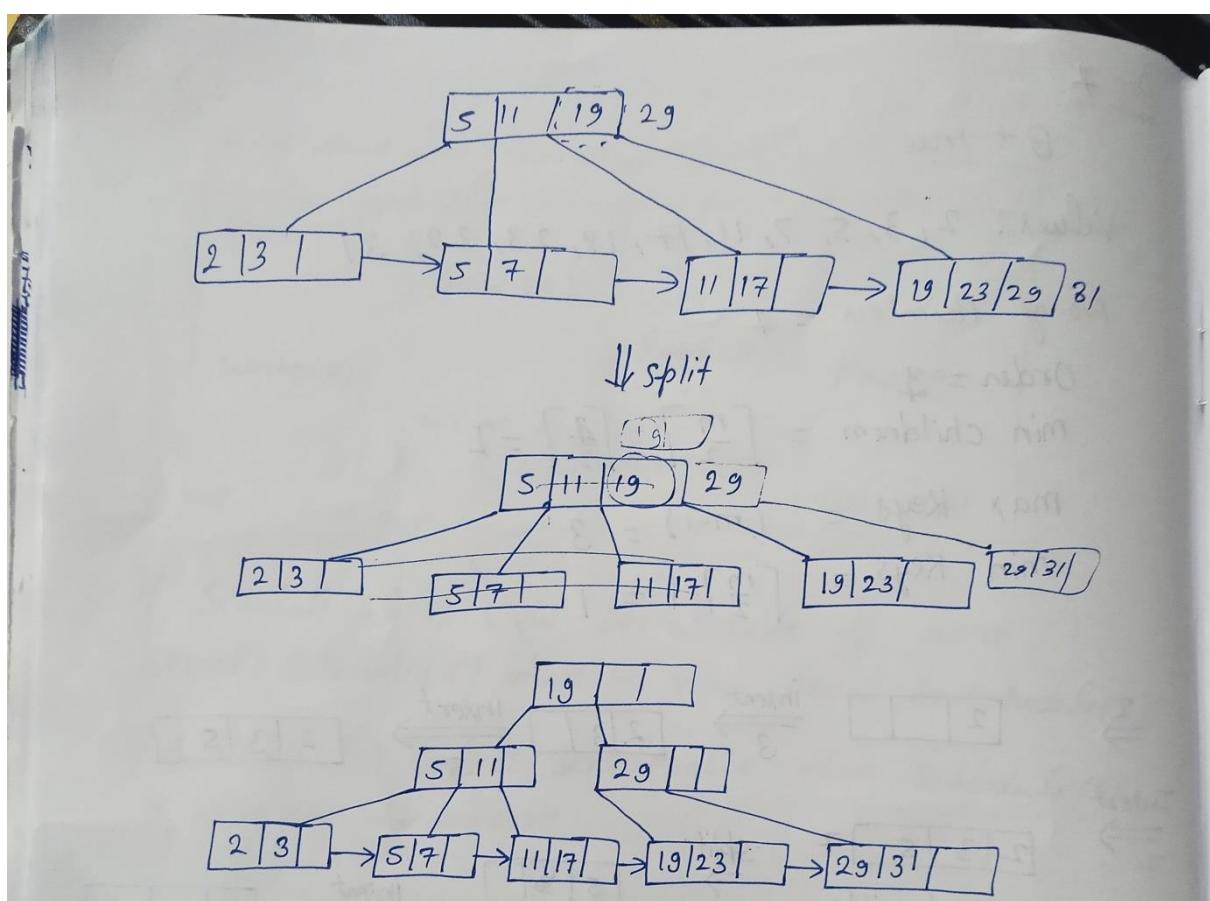
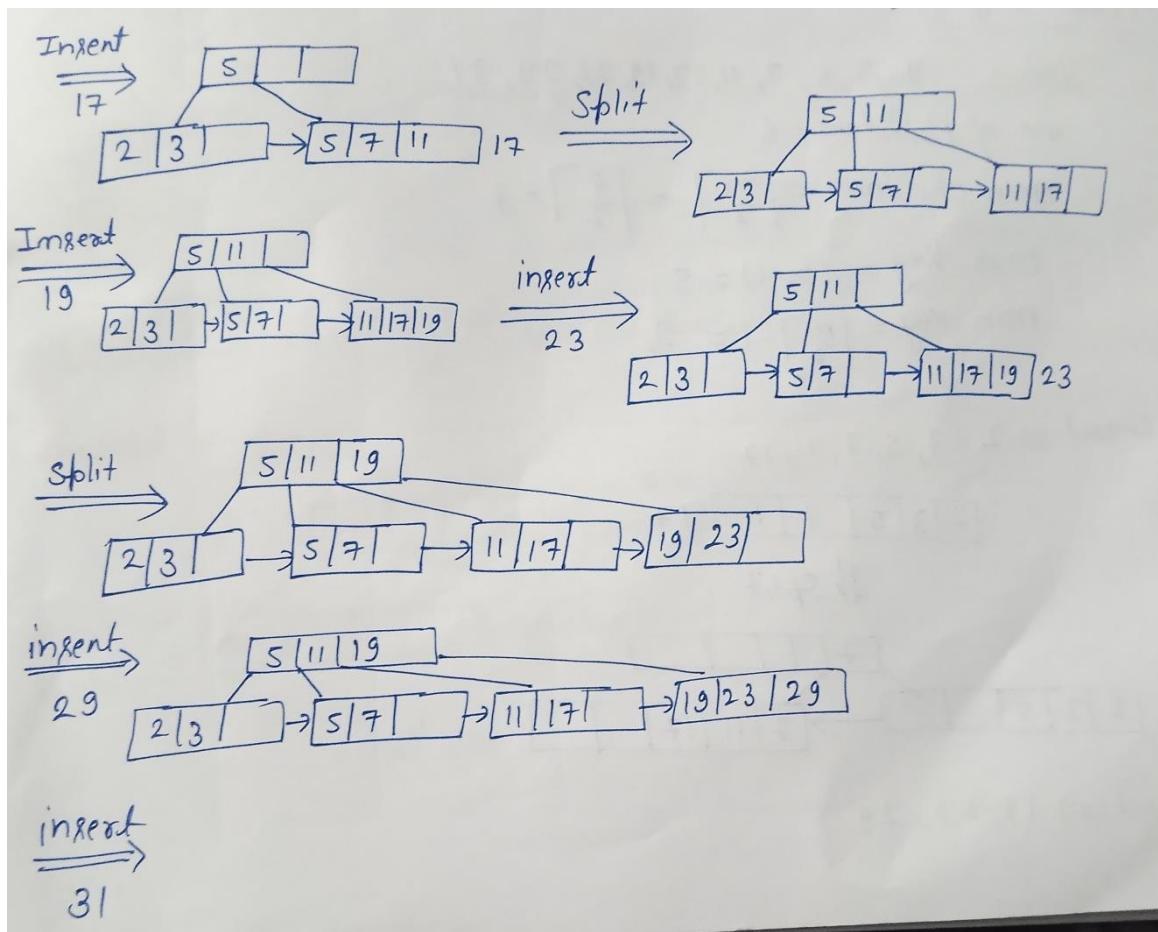
VSS	Given Schedule			View equal Serial Schedule
Final write	A: B : T ₁ T ₂ T ₃ T ₄			(T ₁ T ₂ T ₃) → T ₄
Initial Read	data Item	Initial Read	write	
	A	T ₁		
	B	-	T ₁ T ₂ T ₃ T ₄	T ₁ T ₂ T ₃ → T ₄
Updated Read	-	-	-	-

7. Construct a B+-tree index for the following set of key values: (2, 3, 5, 7, 11, 17, 19, 23, 29, 31). Assume that the tree is initially empty and values are added in ascending order. Construct B+-trees for the cases where the number of pointers that will fit in one node is as follows:

a) Four

Answer :-





b) Six

Answer :-

(b) Order = 6

Values : 2, 3, 5, 7, 11, 17, 19, 23, 29, 31

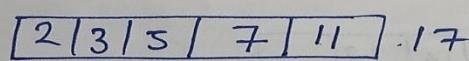
No. of Pointers = 6

$$\text{min children} = \lceil \frac{m}{2} \rceil = \lceil \frac{6}{2} \rceil = 3$$

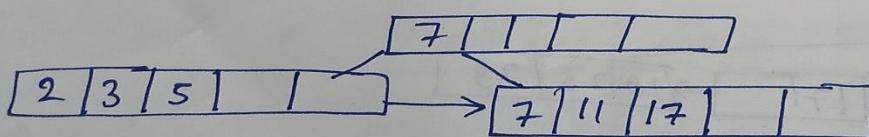
$$\text{max key} = (m-1) = 5$$

$$\text{min key} = \lceil \frac{m}{2} \rceil - 1 = 2$$

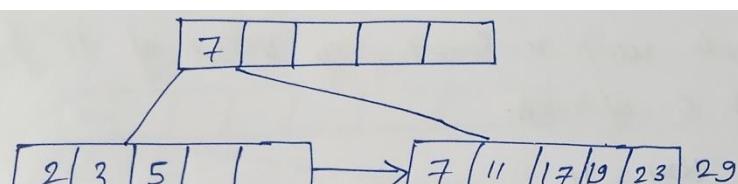
Insert \Rightarrow 2, 3, 5, 7, 11, 17



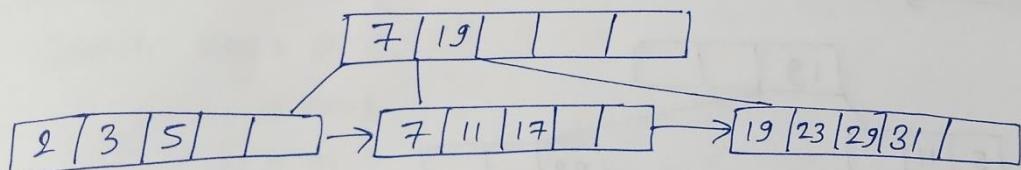
↓↓ split



Insert \Rightarrow 19, 23, 29



↓↓ split



Insert \Rightarrow 31

final free

c) Eight

④ Order = 8

Values : 2, 3, 5, 7, 11, 17, 19, 23, 29, 31

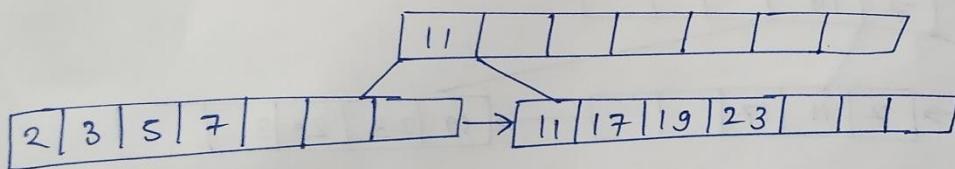
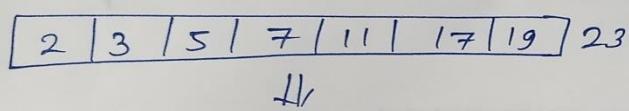
No. of Pointers = 8

min children = $\lceil \frac{m}{2} \rceil = 4$

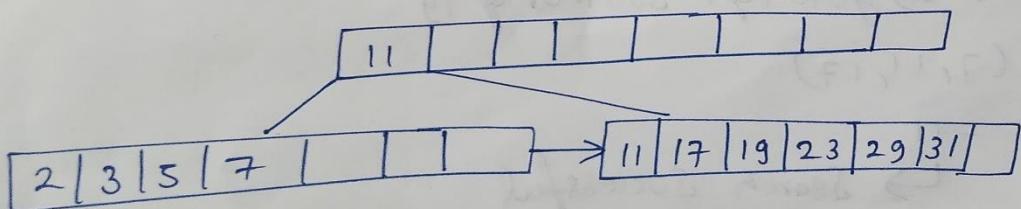
max Key = 7

min Key = 3

Insert \Rightarrow 2, 3, 5, 7, 11, 17, 19, 23

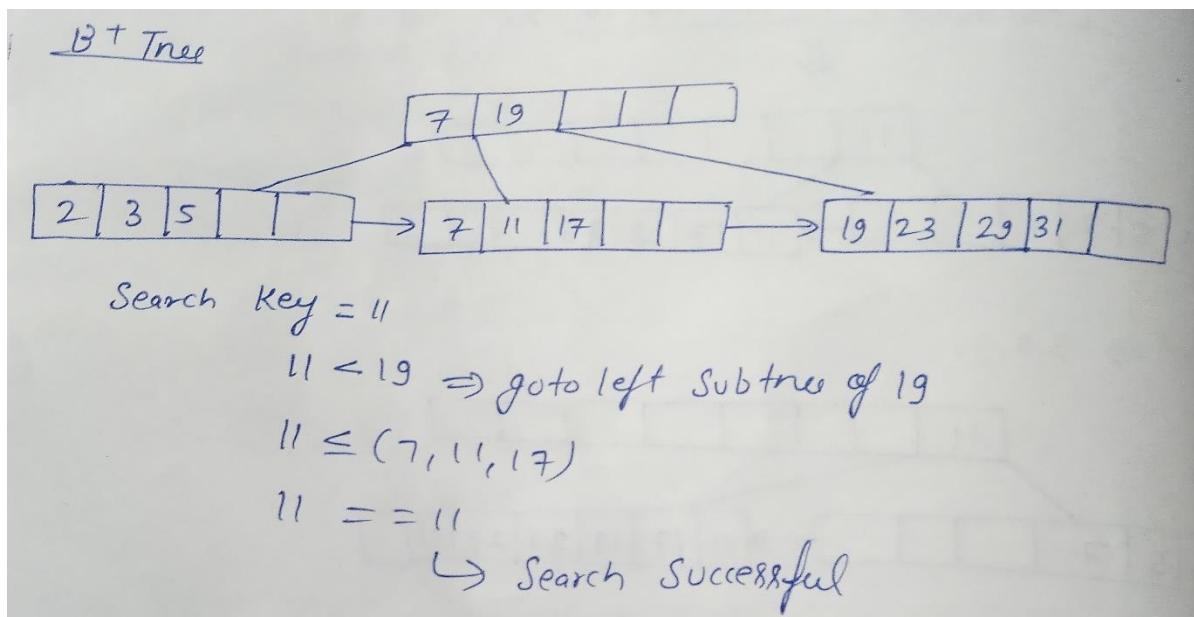
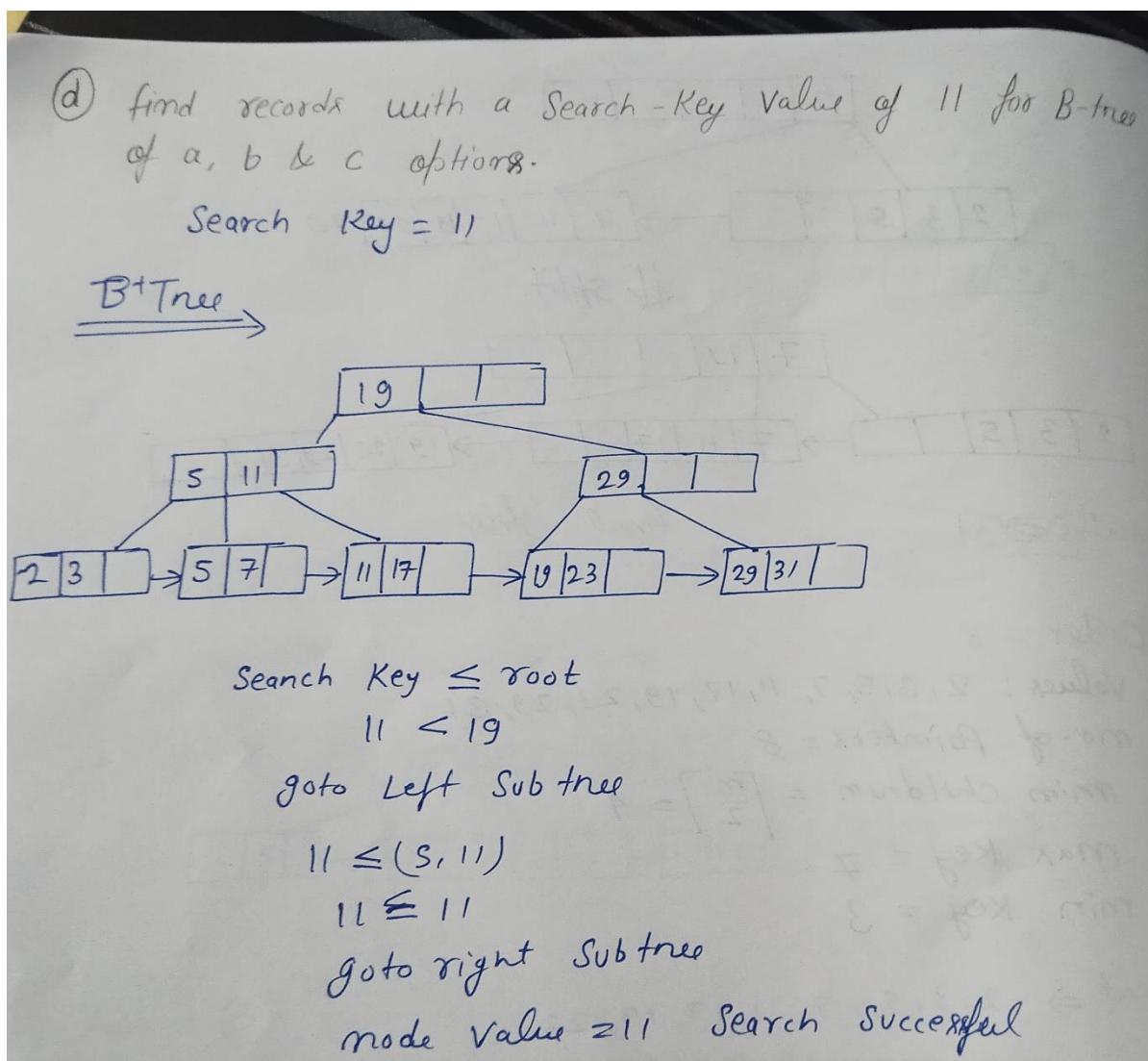


Insert \Rightarrow 29, 31

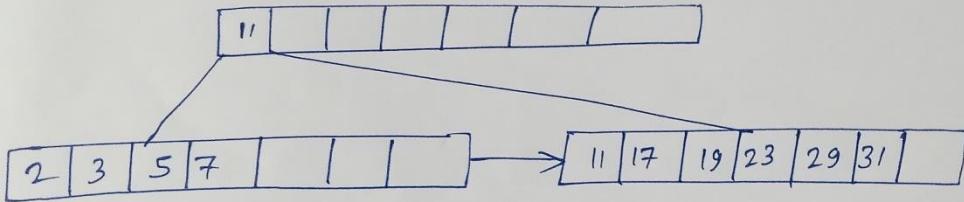


final tree

d) Find records with a search-key value of 11 for B-tree of a, b and c options.



B⁺ tree



Search Key = 11

11 ≤ root

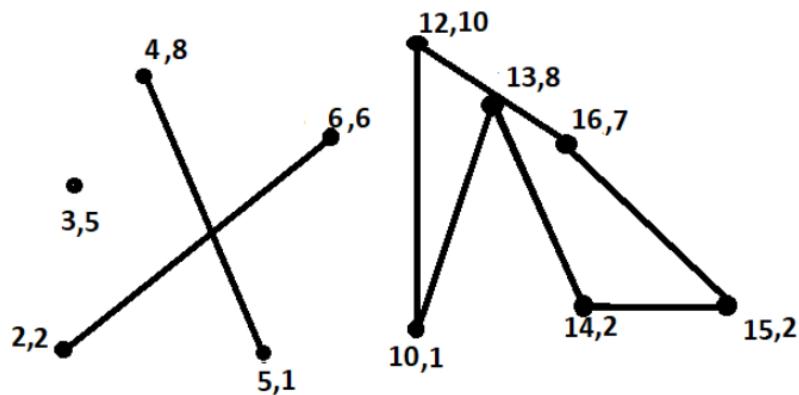
11 ≤ 11

↳ goto right subtree

11 == 11

↳ Search Successful

8. How to record the following object in spatial database. Explain which model is efficient for handing independent objects.

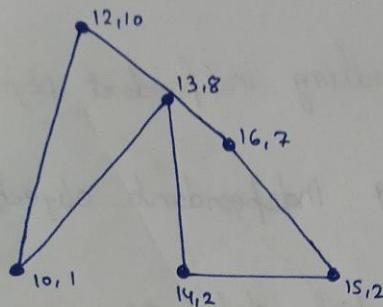


Answer :-

⑧

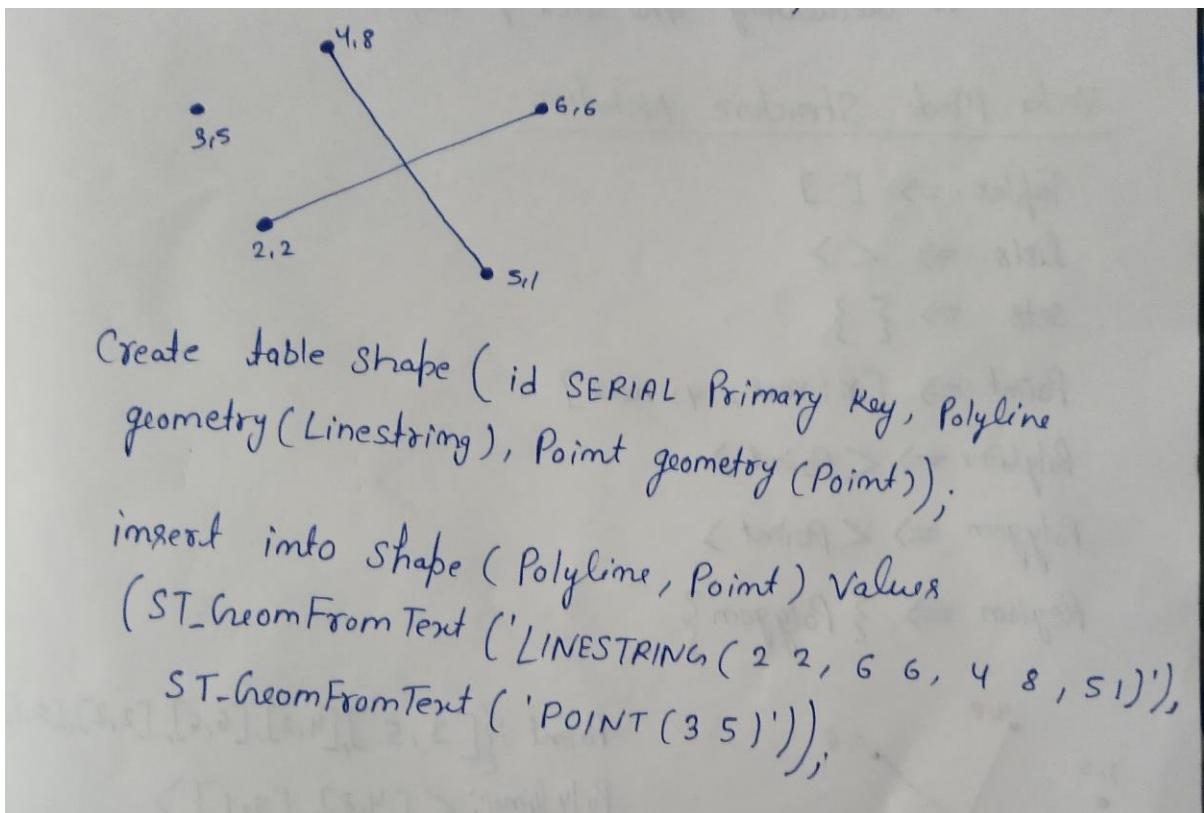
To record the following object in a spatial database we would typically use a geometric model that supports spatial data.

To record the object in the spatial database using the Vector Model, we would store the coordinates of the object's vertices and the connections between them. In this model, each object is stored as a set of coordinates that defines its shape and location.



Create table shapes(id SERIAL Primary Key,
Polygon geometry (Polygon), triangle geometry (Polygon));

insert into shapes (Polygon, triangle) values
(ST_GeomFromText ('POLYGON((13 8, 16 7, 15 2, 14 2,
13 8))'), ST_GeomFromText ('POLYGON(
(10 1, 12 10, 13 8, 10 1))'));



Create table shape (id SERIAL Primary Key, geometry (Linestring), Point geometry (Point)),

insert into shape (Polyline, Point) values
 $(ST_GeomFromText('LINESTRING(2 2, 6 6, 4 8, 5 1)'),$
 $ST_GeomFromText('POINT(3 5)'))$,

which model is efficient for handling independent object

- ⇒ One efficient model for handling independent objects is the vector model.
- ⇒ The vector model represents spatial data as a collection of Points, lines and Polygons.
- ⇒ The vector Model is efficient for handling independent objects because it allows for precise representation and manipulation of geometric data.
- ⇒ It supports various spatial operations such as point-in-Polygon tests, intersection, and buffering.
- ⇒ The model is also suitable for performing spatial queries and analysis, such as finding objects within a certain distance or calculating the area of objects.

Vector Model Structure Notation

Tuples $\Rightarrow []$

Lists $\Rightarrow < >$

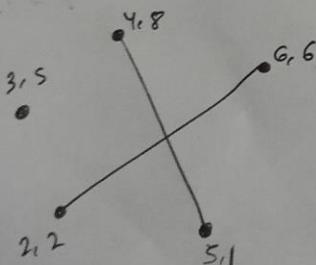
Sets $\Rightarrow \{ \}$

Point $\Rightarrow [x: \text{real}, y: \text{real}]$

Polyline $\Rightarrow < \text{Point} >$

Polygon $\Rightarrow < \text{Point} >$

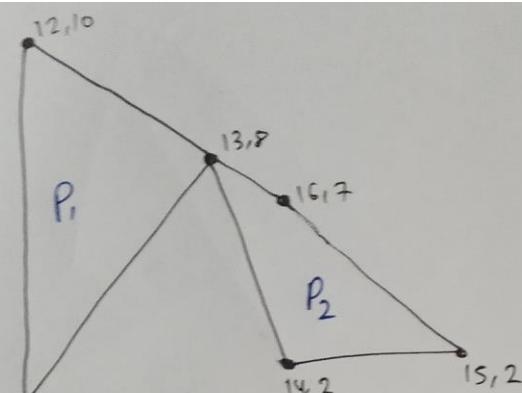
Region $\Rightarrow \{ \text{Polygon} \}$



Point : $\{ [3, 5], [4, 8], [6, 6], [2, 2], [5, 1] \}$

Polyline $L_1 : < [4, 8], [5, 1] >$

Polyline $L_2 : < [6, 6], [2, 2] >$



Points : $\{ [10, 1], [12, 10], [13, 8], [16, 7], [14, 2] \}$

Points : $\{ [10, 1], [12, 10], [13, 8], [16, 7], [15, 2], [14, 2] \}$

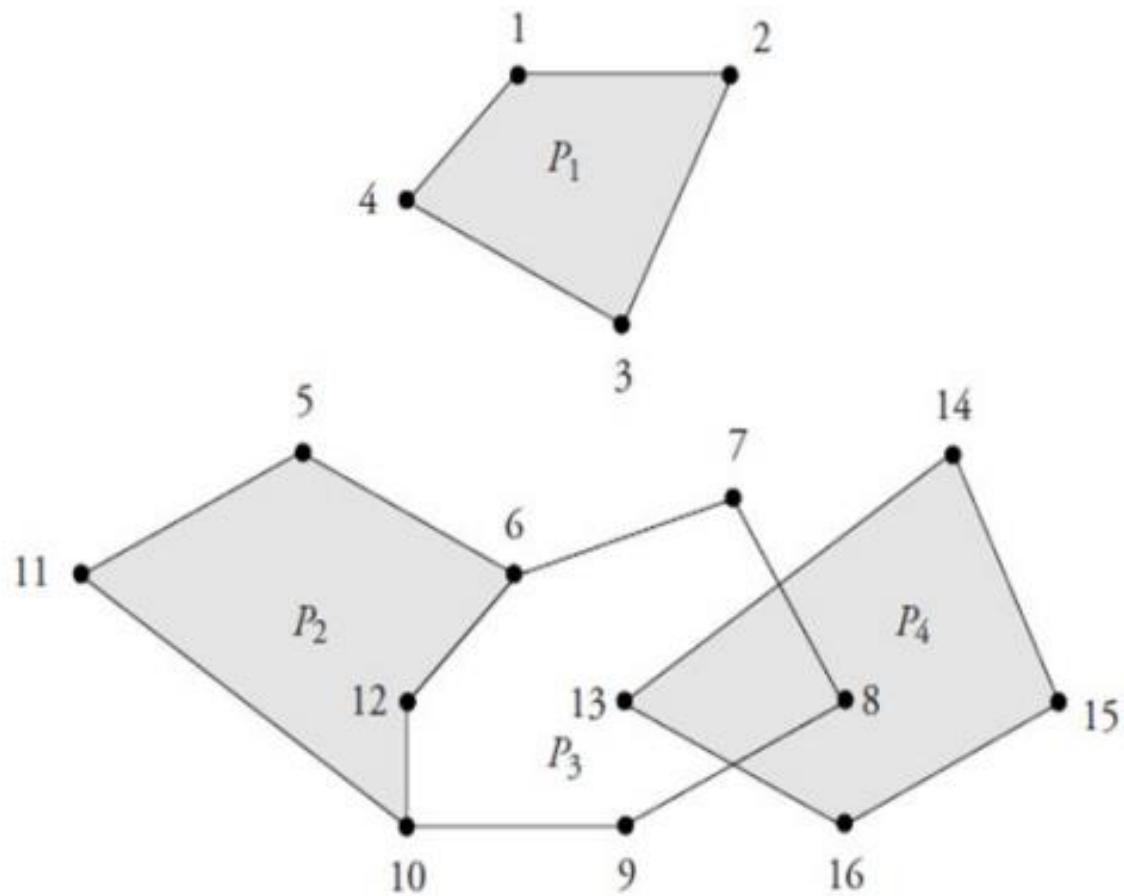
Polyline :
Line

Polygon : $\{ < (10, 1), (12, 10), (13, 8), (10, 1) >, < (13, 8), (16, 7), (15, 2), (14, 2), (13, 8) > \}$

Region : $\{ P_1, P_2 \}$

The Vector model is an efficient choice for handling independent objects in a spatial database due to its precise representation.

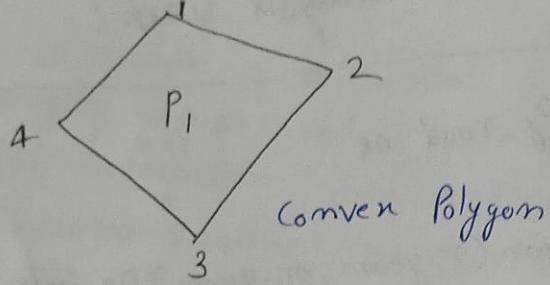
9. Consider the given image is planar, how you would record the objects and suggest a model to handle the objects in planar mode. Explain which representation mode and spatial region ADT operation can used with the model.



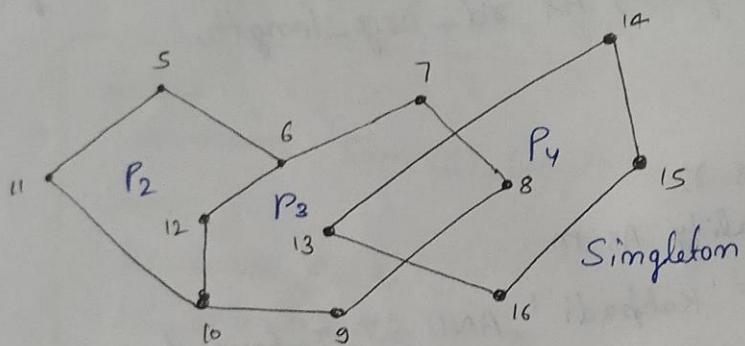
Answer :-

(9)

Vector Representation



Convex Polygon



This is Surface objects (Polygons and regions) with their vector representation, using the vector notation.

The region Consisting of adjacent polygons P_2 and P_3 can be described as:-

$$\{ \langle 5, 6, 12, 10, 11 \rangle, \langle 6, 7, 8, 9, 10, 12 \rangle \}$$

Where each polygon is described by a list of vertices.
The region Consisting of the Single Polygon P_4 is described by the Singleton

$$\{ \langle 13, 14, 15, 16 \rangle \}.$$

Record the object & model to handle the object in Planar mode.
Similarly, the region consisting of polygon P_i
is described as the Singleton $\{ \langle 1, 2, 3, 4 \rangle \}$.

In Summary, this representation allows one to represent a variety of polyline and region objects. Compared to the raster mode, it is a very concise representation, because the memory space required to represent the objects is proportional to the number of vertices of the linear approximation of the objects. The more precise the approximation, the larger the number of points and, of course, the larger the space required for storing the objects.

Again, this representation is loose and cannot check on a large number of constraints. For instance, there is no way to distinguish a simple polygon from a non-simple one, a convex polygon from a non-convex one, a polygon from a polyline, or a set of adjacent polygons from a set of disjoint or intersecting polygons.

Representation Mode

Structure Notation

- ① Tuples $\Rightarrow []$
- ② Lists $\Rightarrow < >$
- ③ Sets $\Rightarrow \{ \}$
- ④ Points $\Rightarrow [x: \text{real}, y: \text{real}]$
- ⑤ Polyline $\Rightarrow < \text{Point} >$
- ⑥ Polygon : $< \text{Point} >$
- ⑦ Region : $\{ \text{Polygon} \}$

LBB

$P_1 : \{1, 2, 3, 4\}$

$P_2 : \{5, 6, 12, 10, 11\}$

$P_3 : \{6, 7, 8, 9, 10, 12\}$

$P_4 : \{13, 14, 15, 16\}$

Sets :-

Adjacent Polygons

$\{P_2, P_3\}$

$\{\{5, 6, 12, 10, 11\}, \{6, 7, 8, 9, 10, 12\}\}$

Region Combinations of the Single polygon

$\{P_4\}$

$\{\{13, 14, 15, 16\}\} \rightarrow \text{Singleton}$

Singleton

$\{P_1\}$

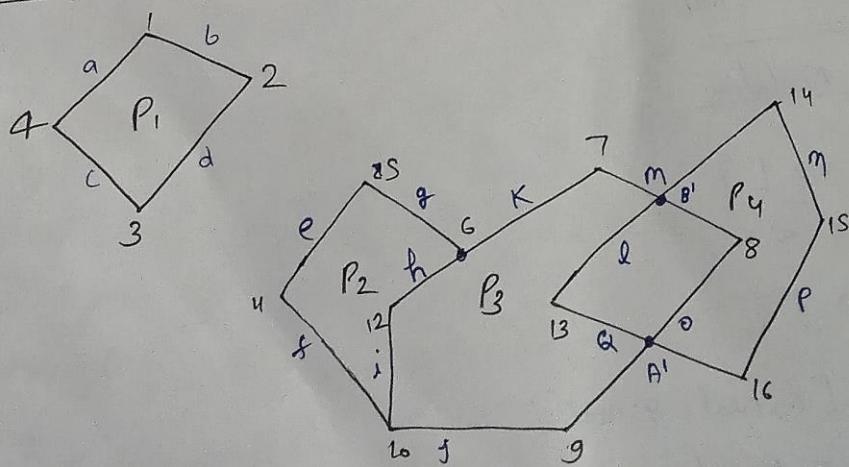
$\{\{1, 2, 3, 4\}\}$

Polygon

Region

$\{P_1, P_2, P_3, P_4\}$

Network Model



Point : $[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16]$

mode : $\{[1, \langle a, b \rangle], [2, \langle b, d \rangle], [3, \langle c, d \rangle], [4, \langle a, c \rangle]\}$
 $\{[1, \langle e, f \rangle], [2, \langle h, i \rangle], [3, \langle j, o \rangle], [4, \langle m, n \rangle],$
 $[5, \langle m, p \rangle], [6, \langle o, p \rangle], [7, \langle e, g \rangle], [8, \langle h, g \rangle], [9, \langle j, o \rangle], [10, \langle f, i \rangle],$
}

arc : $\{[4, 1], [1, 2], [2, 3], [3, 4], [11, 5], [5, 6], [6, 12], [12, 10],$
 $[10, 11], [6, 7], [7, 8], [8, 9], [9, 10], [13, 14], [14, 15], [15, 16],$
[16, 13]

Polygon
 $P_1 \langle 1, 2, 3, 4 \rangle$

$P_2 \langle 5, 6, 12, 10, 11 \rangle$

$P_3 \langle 6, 7, 8, 9, 10, 12 \rangle$

$P_4 \langle 19, 15, 16, 13 \rangle$

Region : $R_1 : \{P_1\}$

$R_2 : \{P_2, P_3\}$

$R_3 : \{P_4\}$

Spatial region ADT operation

① PointInRegion

The ADT opⁿ used to check whether ~~two~~ Region Point belongs to a region.

Region \times Point \rightarrow bool

$R_1 \times (1) \rightarrow$ True

$R_2 \times (1) \rightarrow$ False

② Overlap

Used to test whether two regions intersect.

region \times region \rightarrow bool

$R_1 \times R_2 \rightarrow \text{False}$

$R_2 \times R_3 \rightarrow \cancel{\text{False}} \text{ True}$

③ Intersection

Returns the intersection of two Regions:

region \times region \rightarrow region

$R_1 \times R_2 \rightarrow \text{empty}$

$R_2 \times R_3 \rightarrow \cancel{\text{empty}} R_2 \subset [13, B', 18, A']$

④ Meet

Tests the adjacency of two regions.

region \times region \rightarrow bool

$R_1 \times R_2 \rightarrow \text{False}$

$R_2 \times R_3 \rightarrow \text{True}$

⑤ Area

Return the area size of a region.

region \rightarrow real

⑥ Region Union

{region} \rightarrow region

10. Consider municipality boundaries of Vellore is represented with table of linear roads, and a table of polygonal municipality boundaries.

The definition of the bc_roads table is:

The definition of the bc_roads table is:

Column	Type	Description
gid	integer	Unique ID
name	character varying	Road Name
geom	geometry	Location Geometry (Linestring)

The definition of the bc_municipality table is:

Column	Type	Description
gid	integer	Unique ID

code	integer	Unique ID
name	character varying	City / Town Name
Geom	Geometry	Location Geometry

Give the spatial queries for given statement:

a) What is the total length of all roads, expressed in kilometers?

① Select Sum(ST_Length(geom))/1000 As Km_roads
From bc_roads;

b) How large is the city of Katpadi, in hectares?

(b) This query combines an attribute condition (on the municipality name) with a spatial calculation (of the area):

```
Select ST_Area(geom)/10000 As hectares  
from bc_municipality  
where name = 'Katpadi';
```

c) What is the largest municipality in the province, by area?

(c) This query brings a spatial measurement into the query condition. There are several way of approaching this problem, but the most efficient is below:

```
Select name, ST_Area(geom)/10000 As hectares  
from bc_municipality  
Order By hectares DESC  
LIMIT 1;
```

In order to answer this query we have to calculate the area of every polygon. If we were doing this a lot it would make sense to add an area column to the table that we could separately index for performance. By ordering the results in a descending direction, and then using the PostgreSQL "LIMIT" command we can easily pick off the largest value without using an aggregate function like max().

d) What is the length of roads fully contained within each municipality?

④ This is an example of a "spatial join", because we are bringing together data from two tables but using a spatial interaction condition ("Contained") as the join condition rather than the usual relational approach of joining on a common key:

```
Select m.name, Sum(ST_Length(r.geom))/1000 as roads_km  
from bc_roads As r,  
bc_municipality As m  
where ST_Contains(m.geom, r.geom)  
Group By m.name  
ORDER BY roads_km;
```

This query takes a while, because every road in the table is summarized into the final result. For smaller overlays the response can be very fast.

e) Create a new table with all the roads within the city of Katpadi.

(e) This is an example of an "overlay", which takes in two tables and outputs a new table that consists of spatially clipped or cut resultants. Unlike the "spatial join" demonstrated above, this query actually creates new geometries. An overlay is like a turbo-charged spatial join, and is useful for more exact analysis work:

Create table Pg_road as

Select

ST_Intersection(r.geom, m.geom) As intersection_geom,
ST_Length(r.geom) As rd_orig_length,
r.*

From

bc_roads As r,

bc_municipality As m

where m.name = 'Katpadi' AND ST_Intersects(r.geom,
m.geom);

f) What is the length in kilometers of "VIT" in Vellore?

(f) Select
 Sum(ST_Length(r.geom)) /1000 As Kilometers
 from
 bc_roads r,
 bc_municipality m
 Where r.name = 'VIT' AND m.name = 'Vellore'
 AND ST_Contains(m.geom, r.geom),

g) What is the largest municipality polygon that has a hole?

(g) Select gid, name, ST_Area(geom) As area
 from bc_municipality
 Where ST_NRings(geom) > 1
 ORDER BY area DESC LIMIT 1;

11. Consider the spatial database with following schema.

- **Country** (country_code integer, country_name varchar (30), geometry region, Primary Key (country_code))
- **State** (state_code integer, state_name varchar (30), country_code varchar (30), geometry region, Primary Key (state_code), Foreign Key (country_code) References Country)
- **County** (county_code integer, county_name varchar (30), state_code varchar (30), population integer, geometry region, Primary Key (county_code), Foreign Key (state_code) References State)
- **Highway** (highway_code integer, highway_name varchar (4), highway_type varchar (2), Primary Key (highway_code))
- **HighwaySection** (section_code integer, section_number integer, highway_code integer, Primary Key (section_code,highway_code), Foreign Key (section_code) References Section, Foreign Key (highway_code) References Highway)
- **Section** (section_code integer, section_name varchar (4), number_lanes integer, city_start varchar (30), city_end varchar (30), geometry_line, Primary Key (section_code), Foreign Key (city_start) References City, Foreign Key (city_end) References City)

- **City** (city_name varchar (30), population integer, geometry_region, Primary Key (city_name))
- **Land Use** (region_name varchar (30), land_use type varchar (30), geometry region, Primary Key (region_name))

a) Number of inhabitants in the County of San Francisco.

Ans-11

① Number of inhabitants in the County of San Francisco.

Select Population
from County
where County_name = 'San Francisco'

b) List of the names of the counties in the State of California.

② List of Counties of the State of California.

Select County_name
from County, State
from State.State_Code = County.State_Code
and State_name = 'California'

c) Number of inhabitants of the United States of America.

⑥ Number of inhabitants of the US of America.
Select Sum(c2.population)
from Country c1, State s, County c2
Where c1.country-name = 'USA'
and c1.country-code = s.country-code
and s.state-code = c2.state-code

d) Number of lanes in the first section of Interstate 99.

⑦ Number of lanes in the first Section of Interstate 99.
Select s.number-lanes
from Highway h1, HighwaySection h2, Section s
where h1.highway-code = h2.highway-code
and h2.Section-Code = s.Section-Code
and h1.highway-name = 'I99'
and h2.section-number = 1

e) Names of sections that constitute Interstate 99.

(e) Name of Sections that constitute Interstate 99.

Select S.Section_name
from Highway h₁, HighwaySection h₂, Section S
where h₁.highway_name = 'I99'
and h₁.highway_code = h₂.highway_code
and h₂.Section_code = S.Section_Code

f) Counties adjacent to the County of San Francisco in the same state.

(f) Counties adjacent to the County of San Francisco in the same state.

Select C₁.county_name
from County C₁, County C₂
where C₂.county_name = 'San Francisco'
and C₁.state_code = C₂.state_code
and ST_Touches(C₁.geometry, C₂.geometry)

g) Counties larger than the largest county in California.

(g) Counties larger than the largest county in California.

Select C1.County_name
from County C1
where ST_Area(C1.geometry) >
(Select max(ST_Area(C1.geometry))
from County C1, State S
where S.State_code = C1.State_code
and S.State_name = 'California')

h) Length of Interstate 99.

(h) Length of Interstate 99.

Select sum(ST_Length(S.geometry))
from Highway h1, HighwaySection h2, Section S
where h1.highway_name = 'I99'
and h1.highway_code = h2.highway_code
and h2.Section_Code = S.Section_code

i) All highways going through the State of California.

⑩ All highways going through the state of California
Select distinct h1.highway_name
from State s1, Highway h1, HighwaySection h2, Section s2
where s1.state_name = 'California'
and h1.highway_code = h2.highway_code
and h2.section_code = s2.section_code
and ST_Overlaps(s2.geometry, s1.geometry)

j) Display of all residential areas in the County of San Jose.

(j) Display of all residential areas in the County of San Jose.
Select ST_Intersection(l.geometry, c.geometry)
from County c, LandUse l
where c.County_name = 'San Jose'
and l.land_use_type = 'residential area'
and ST_Overlaps(l.geometry, c.geometry)

k) Overlay of the themes of counties and land use.

(k) Overlay of the themes of counties and land use.
Select County_name, land_use_type, ST_Intersection(c.geometry, l.geometry)
from County c, LandUse l
where ST_Overlaps(c.geometry, l.geometry)

l) Description of the county pointed to on the screen.

① Description of the County Pointed to on the Screen.

Select County_name, Population
from County

Where ST_Contains(geometry, @point)

m) Counties that intersect a given rectangle on the screen.

② Counties that intersect a given rectangle on the Screen.

Select County_name
from County

Where ST_Overlaps(geometry, @rectangle)

n) Parts of the counties that are within a given rectangle on the screen.

③ Part of the Counties that are within a given rectangle on the Screen.

Select ST_Intersection(geometry, @rectangle)
from County

Where ST_Overlaps(geometry, @rectangle)

o) Description of the section pointed to on the screen.

(O) Description of the section pointed to on the screen.

Select section name, number_lanes
from section

where ST_Contains(geometry, @Point)

p) Description of the highway(s) of which a section is pointed to on the screen.

(P) Description of the highway(s) of which a section is pointed to on the screen.

Select h1.highway_name, h1.highway_type
from Highway h1, HighwaySection h2, Section S
where h1.highway_code = h2.highway_code
and h2.Section_Code = S.Section_Code
and ST_Contains(S.geometry, @Point)

12. Consider the relations employee (name, address, salary, plant-number), machine (machine-number, type, plant-number). Assume that the employee relation is fragmented horizontally by plant-number, and that each fragment is stored locally at its corresponding plant site. Assume that the machine relation is stored in its entirety at the Chennai site. Describe a good strategy for processing each of the following queries.

a) Find all employees at the plant that contains machine number 1130.

(12) (a)

(i)- Perform $\pi_{\text{Plant number}}(\sigma_{\text{Machine number} = 1130} \text{ machine})$ at Armonk.

(ii)- Send the query $\pi_{\text{Name (Employee)}}$ to all site(s) which are in the result of the previous query.

(iii)- Those sites compute the answers.

(iv)- Union the answers at the destination site.

Retrieve Plant number for machine number 1130

Select plant number

from machine

where machine_number = 1130;

Query the employee relation locally at the plant Site.

Select *

from employee

where plant_number = < plant_number >;

b) Find all employees at plants that contain machines whose type is "millingmachine."

(b)

This strategy is the same as (a), except the first step should be to perform

$\pi_{\text{Plant number}}(\sigma_{\text{type} = \text{"milling machine"} (\text{Machine})})$ at Armonk.

(b) # Retrieve plant numbers for machines of types "milling machine" from the machine relation stored at chennai site

Select distinct Plant_number
from machine

Where type = 'milling machine';

Query the employee relation locally at each plant site.

Select *

from employee

where plant_number IN (<plant_number_1>,
<plant_number_2>, ...);

c) Find all machines at the Mumbai plant.

(c)

(i)- Perform

Plant number = x (machine) at Almaden,
Where x is the plant number for Almaden

(ii)- Send the answers to the destination site.

Query the machine relation stored at chennai site to retrieve all machines at the Mumbai plant

Select *

from machine

Where plant_number = <Mumbai_plant_number>;

d) Find employee _ machine.

(d) Strategy 1

- (i)- Group machine at Armork by plant number.
- (ii)- Send the groups to the sites with the corresponding plant number.
- (iii)- Perform a local join between the local data and the received data.
- (iv)- Union the results at the destination site.

Strategy 2

Send the machine relation at Armork, and all the fragments of the employee relation to the destination site.

For Academic Purposes Only

Select *

from employee e

JOIN machine m ON e.plant_number = m.plant_number;

13. Consider the schema r and s given below

13. Consider the schema r and s given below

SCHEMA r		
A	B	C
1	2	3
4	5	6
1	2	4
5	3	3
8	9	7

SCHEMA s		
C	D	E
3	4	5
3	6	8
2	3	2
1	4	1
1	2	3

What is the output of $r \bowtie s$ and $s \bowtie r$?

(13)

$r \bowtie s$
Left Semijoin

$\bowtie \rightarrow$ Left Semijoin

$r \bowtie s$

$s \bowtie r$

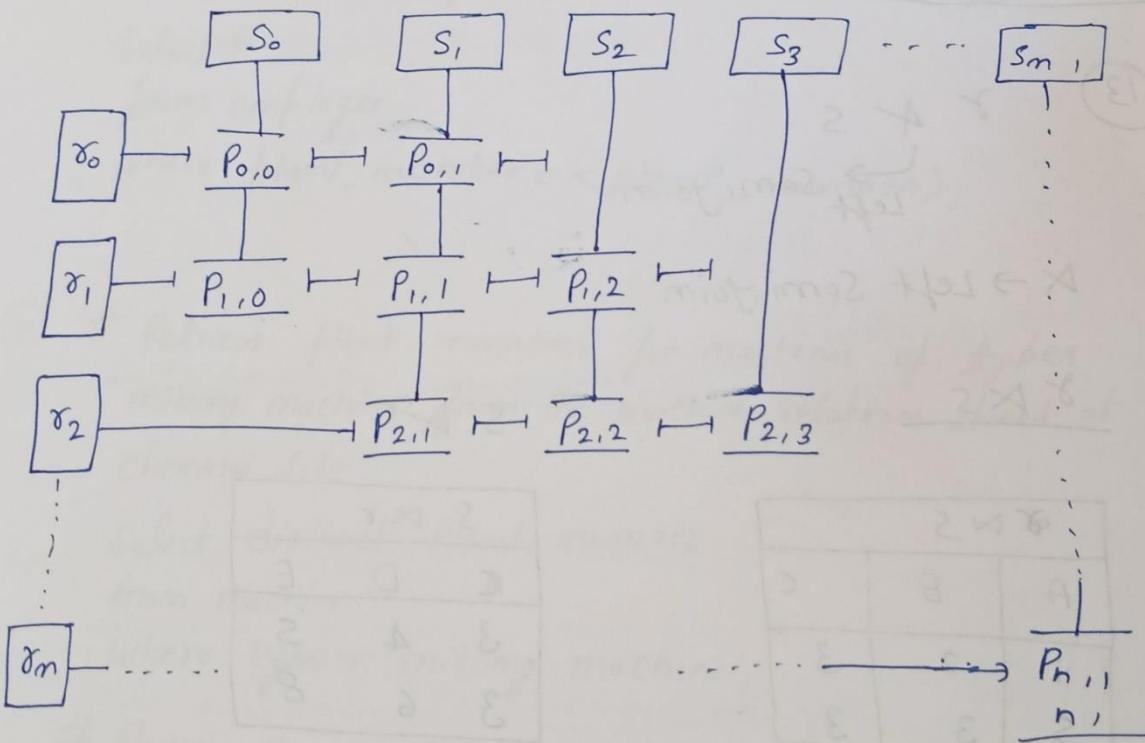
$r \bowtie s$		
A	B	C
1	2	3
5	3	3

$s \bowtie r$		
C	D	E
3	4	5
3	6	8

14. Consider join processing using symmetric fragment and replicate with range partitioning. How can you optimize the evaluation if the join condition is of the form $|r.A - s.B| \leq k$, where k is a small constant. Here, $|x|$ denotes the absolute value of x .

(14)

Relation r is partitioned into m partitions, r_0, r_1, \dots, r_{m-1} , and s is also partitioned into m partitions, s_0, s_1, \dots, s_{m-1} . The partitions are replicated and assigned to processors as shown below.



Each fragment is replicated on 3 processors only, unlike in the general case where it is replicated on n processors. The number of processors required is now approximately $3n$, instead of n^2 in the general case. Therefore given the same number of processors, we can partition the relations into more fragments w.r.t with this optimization, thus making each local join faster.

15. Consider histograms are used for constructing load-balanced range partitions.

a) Suppose you have a histogram where values are between 1 and 100, and are partitioned into 10 ranges, 1 – 10, 11 – 20, . . . , 91 – 100, with frequencies 15, 5, 20, 10, 10, 5, 5, 20, 5, and 5, respectively. Give a load-balanced range partitioning function to divide the values into 5 partitions.

(15) (a) A partitioning vector which gives 5 partitions with 20 tuples in each partition is :-
[21, 31, 51, 76].

The 5 partitions obtained are in arriving at this partitioning vector is that within a histogram range, each value is equally likely.

b) Write an algorithm for computing a balanced range partition with p partitions, given a histogram of frequency distributions containing n ranges.

(b) Let the histogram ranges be called h_1, h_2, \dots, h_k , and the partitions P_1, P_2, \dots, P_p . Let the frequencies of the histogram ranges be m_1, m_2, \dots, m_k . Each partition should contain N/p tuples, where,

$$N = \sum_{i=1}^k m_i$$

To construct the load balanced partitioning vector, we need to determine the value of the K_1^{th} tuple, the value of the K_2^{th} tuple and so on, where

$$K_1 = N/p, K_2 = 2N/p \text{ etc,}$$

until K_{p-1} . The partitioning vector will then be $[K_1, K_2, \dots, K_{p-1}]$. The value of the K_i^{th} tuple is determined as follows. First determine the histogram range h_j in which it falls.

Assuming all values in a range are equally likely, the K_i^{th} value will be

$$\boxed{S_j + (e_j - s_j) * \frac{K_{ij}}{m_j}}$$

Where:

$s_j \rightarrow$ First value in h_j

$e_j \rightarrow$ Last value in h_j

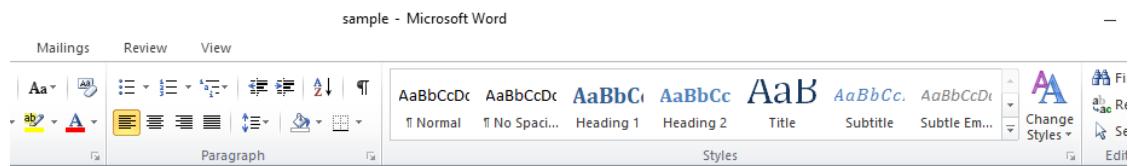
$K_{ij} \rightarrow K_i - \sum_{l=1}^{j-1} m_l$

16. Perform the following tasks

- Create a sample document in Microsoft Word containing this exercise with about the
- same formatting, and save it as **sample.doc**.

Answer :-

My document should look like the exercise.



Exercise : One document, several XML representations

- Perform the following tasks
- Create a sample document in Microsoft Word containing this exercise with about the
- same formatting, and save it as **sample.doc**.
- Copy the contents (as raw text) of **sample.doc** into the editor and specify its structure using XML tags. (Title, sections). Also mark specific styles (such as italic) inside the text.
- Save it as **doc1.xml**.
- Open **sample.doc** in Word and save it as XML (Save as... **XML Document (*.xml)**) with the name **doc2.xml**.
- Open **sample.doc** in **OpenOffice** and save it in the **OpenOffice** format. Change the
- extension of the file from .write to .zip. Extract from the zip file the **content.xml** file and rename it to **doc3.xml**.

- Copy the contents (as raw text) of **sample.doc** into the editor and specify its structure using XML tags. (Title, sections). Also mark specific styles (such as italic) inside the text.
- Save it as **doc1.xml**.

The screenshot shows the Oxygen XML Editor interface. The title bar says 'doc1.xml [C:\Users\god\Music\doc1.xml] - <oXygen/> XML Editor (Academic use only)'. The menu bar includes 'Edit', 'Find', 'Project', 'Options', 'Tools', 'Document', 'Window', and 'Help'. The toolbar has icons for file operations, search, and XML processing. The left pane shows an outline of the XML structure with sections for 'document', 'title', and 'section'. The right pane shows 'Transformation Scenarios - doc1.xml' with two scenarios: 'Flowers sample to WebHelp Respo...' and 'Garage sample to WebHelp Respo...'. The main pane displays the XML code:

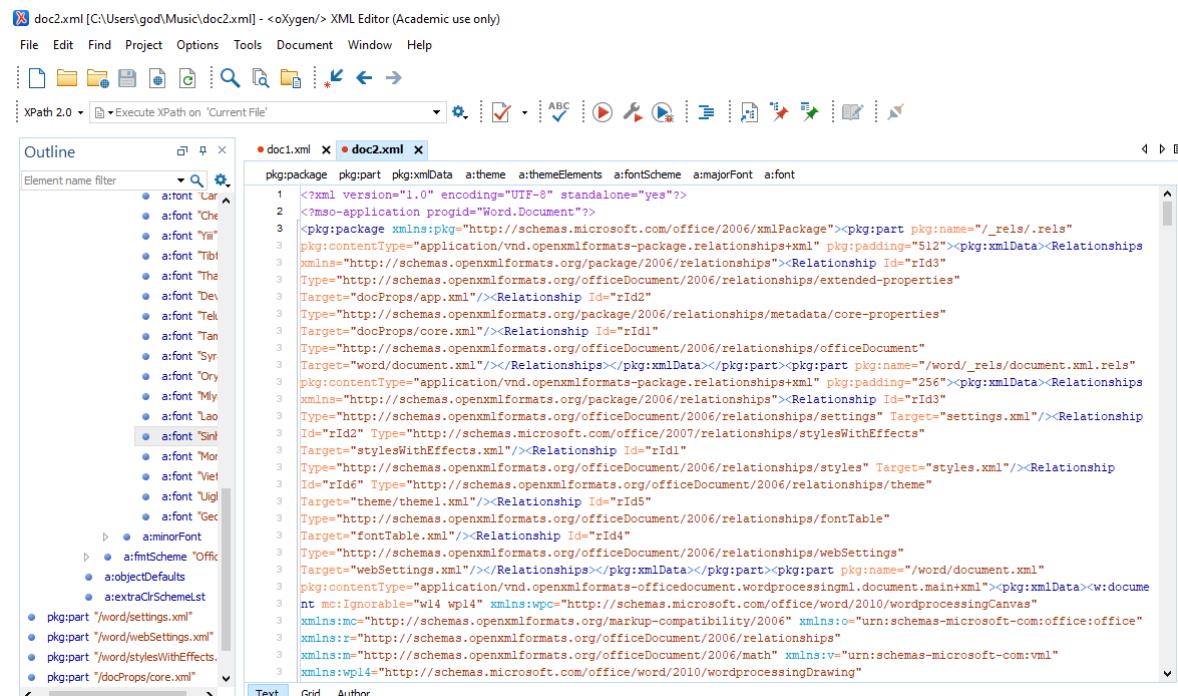
```
<?xml version="1.0" encoding="UTF-8"?>
<document Exercise : One document, several XML representations>
  <title>Exercise : One document, several XML representations</title>
  <section nr="1">16. Perform the following tasks
    <subsection>a) Create a sample document in Microsoft Word containing
      this exercise with about the</subsection>
    <subsection>b) same formatting, and save it as
      <b>sample.doc</b></subsection>
    <subsection>c) Copy the contents (as raw text) of <b>sample.doc</b> in
      oXygen and specify its structure using XML tags. (Title, sections). Also
      mark specific styles (such as italic) inside the text. </subsection>
    <subsection>d) Save it as
      <b>doc1.xml</b></subsection>
    <subsection>e) Open <b>sample.doc</b> in Word and save it as XML (Save
      as... XML Document (*.xml)) with the name <b>doc2.xml</b>.</subsection>
    <subsection>f) Open <b>sample.doc</b> in OpenOffice and save it in the
      OpenOffice format. Change the</subsection>
    <subsection> extension of the file from .write to .zip.
      Extract from the zip file the <b>content.xml</b> file and rename it to
      <b>doc3.xml</b>.</subsection>
  </section>
</document>
```

```

<?xml version="1.0" encoding="UTF-8"?>
<document>
  <title>Exercise : One document, several XML representations</title>
  <section nr="1">16. Perform the following tasks
    <subsection>a) Create a sample document in Microsoft Word containing
      this exercise with about the</subsection>
      <subsection>b) same formatting, and save it as
        <b>sample.doc</b>.</subsection>
      <subsection>c) Copy the contents (as raw text) of <b>sample.doc</b> in
        oXygen and specify its structure using XML tags. (Title, sections). Also
        mark specific styles (such as italic) inside the text. </subsection>
      <subsection>d) Save it as
        <b>doc1.xml</b>.</subsection>
      <subsection>e) Open <b>sample.doc</b> in Word and save it as XML (Save
        as... XML Document (*.xml)) with the name <b>doc2.xml</b>.</subsection>
      <subsection>f) Open <b>sample.doc</b> in OpenOffice and save it in the
        OpenOffice format. Change the</subsection>
        <subsection>extension of the file from .write to .zip.
        Extract from the zip file the <b>content.xml</b> file and rename it to
        <b>doc3.xml</b>.</subsection>
      </section>
    </document>

```

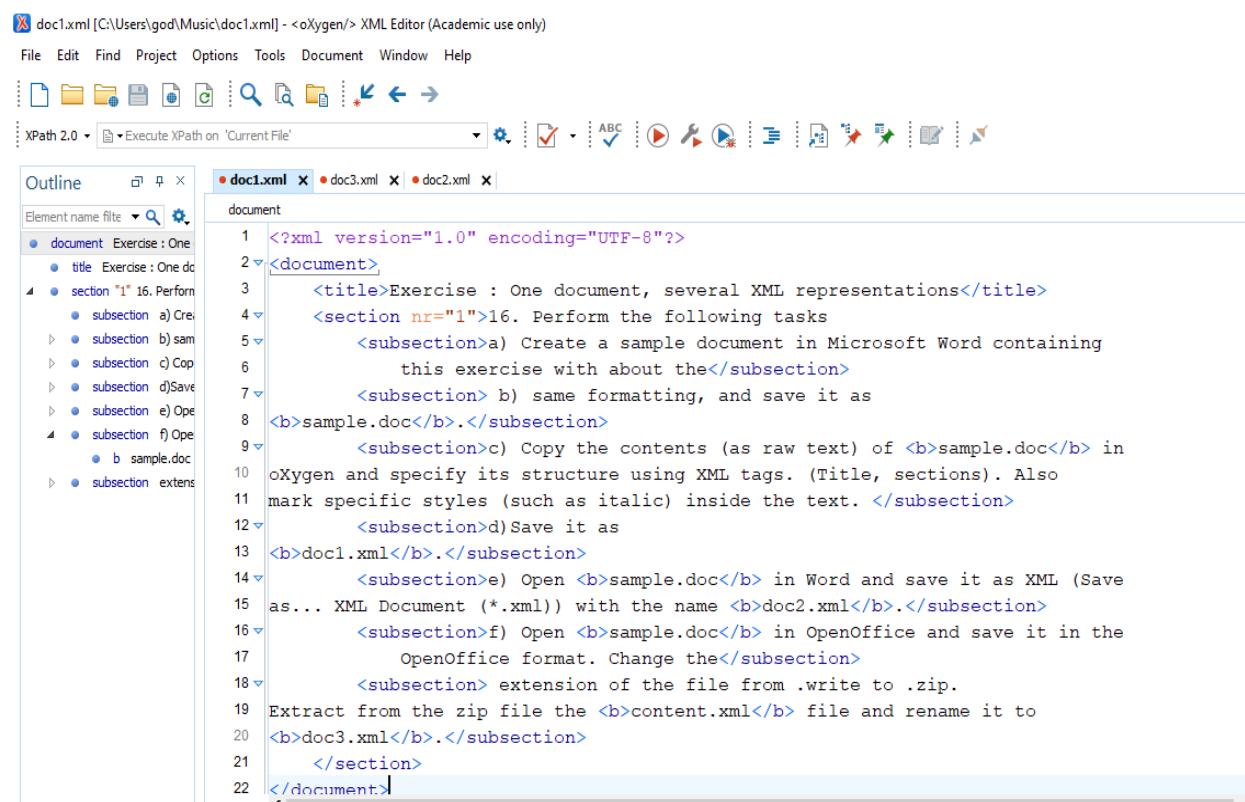
e) Open sample.doc in Word and save it as XML (Save as... XML Document (*.xml)) with the name doc2.xml.



- f) Open sample.doc in OpenOffice and save it in the OpenOffice format. Change the
 g) extension of the file from .write to .zip. Extract from the zip file the content.xml file and
 rename it to doc3.xml.

Name	Date modified	Type	Size
doc1	30-05-2023 00:19	XML Document	2 KB
doc2	30-05-2023 00:22	XML Document	56 KB
doc3	30-05-2023 00:50	XML Document	2 KB
sample	30-05-2023 00:17	Microsoft Word D...	14 KB
sample	30-05-2023 00:39	OpenDocument T...	17 KB
sample	30-05-2023 00:40	WinRAR ZIP archive	15 KB

Doc1.xml file view:-



```

doc1.xml [C:\Users\god\Music\doc1.xml] - <oXygen/> XML Editor (Academic use only)
File Edit Find Project Options Tools Document Window Help
Outline Element name filter XPath 2.0 Execute XPath on 'Current File' ABC
document
  1 <?xml version="1.0" encoding="UTF-8"?>
  2 <document>
  3   <title>Exercise : One document, several XML representations</title>
  4   <section nr="1">16. Perform the following tasks
  5     <subsection>a) Create a sample document in Microsoft Word containing
  6       this exercise with about the</subsection>
  7     <subsection>b) same formatting, and save it as
  8     <b>sample.doc</b>.</subsection>
  9     <subsection>c) Copy the contents (as raw text) of <b>sample.doc</b> in
 10    oXygen and specify its structure using XML tags. (Title, sections). Also
 11    mark specific styles (such as italic) inside the text. </subsection>
 12     <subsection>d) Save it as
 13     <b>doc1.xml</b>.</subsection>
 14     <subsection>e) Open <b>sample.doc</b> in Word and save it as XML (Save
 15 as... XML Document (*.xml)) with the name <b>doc2.xml</b>.</subsection>
 16     <subsection>f) Open <b>sample.doc</b> in OpenOffice and save it in the
 17       OpenOffice format. Change the</subsection>
 18     <subsection> extension of the file from .write to .zip.
 19 Extract from the zip file the <b>content.xml</b> file and rename it to
 20 <b>doc3.xml</b>.</subsection>
 21   </section>
 22 </document>

```

Doc2.xml file view :-

The screenshot shows the oXygen XML Editor interface. The main window displays the XML code for 'doc2.xml' with several relationships defined. The 'File' menu includes 'File', 'Edit', 'Find', 'Project', 'Options', 'Tools', 'Document', 'Window', and 'Help'. A toolbar at the top includes icons for file operations like Open, Save, Print, and Find. Below the toolbar is the 'XPath 2.0' search bar and the 'Execute XPath on 'Current File'' dropdown. On the left is the 'Outline' panel with an 'Element name filter' and a tree view of the XML structure. On the right is the 'Entities' panel with an 'Entity filter' and a table for managing entity names and values. The central area contains the XML code for 'doc2.xml'.

Doc3.xml file view :-

doc3.xml [C:\Users\god\Music\doc3.xml] - <oXygen/> XML Editor (Academic use only)

File Edit Find Project Options Tools DocBook4 Document Window Help

XPath 2.0 Execute XPath on 'Current File'

Outline Element name filter

para

para Exercise : One document, several XML representations

16. Perform the following tasks

a) Create a sample document in Microsoft Word containing this exercise with about the same formatting, and save it as sample.doc.

b) Copy the contents (as raw text) of sample.doc into the editor and specify its structure using XML tags. (Title, sections). Also mark specific styles (such as italic) inside the text.

c) Save it as doc1.xml.

d) Open sample.doc in Word and save it as XML (Save as... XML Document (*.xml)) with the name doc2.xml.

e) Open sample.doc in OpenOffice and save it in the OpenOffice format. Change the extension of the file from .write to .zip. Extract from the zip file the content.xml file and rename it to doc3.xml.

para

Entities

Name	Value
lt	<
gt	>
amp	&
apos	'
quot	"
euro	€

17. Correct the following XML document to be well-formed (hint: use the XML editor):

```
<?xml version="1.0" encoding="utf-16"?>
<movies>
    <movie id="56225">
        <title>Love Story</title>
        <title></title>
        <year>1980</year>
        <_director name='Coppola'></_director>
        <comment text="Five start" text="Average"/>
        <xml>Introduce XML content</xml>
        <newcomment text="An <important> text">Oscar</newcomment>
        <comment lang=de>&copy; 1980 Warner Bros.</comment>
        <!-- Famous movie of the --80s -->
    </Movie>
</movies>
```

Answer :-

(17)

This document has the following Problems:-

- ⇒ The quotes in XML must always be simple quotes or double quotes, but not "word-style" quotes ("or..")
- ⇒ The movie start tag does not correspond to the Movie end tag.
- ⇒ The entity © is not defined in XML. Some XML-based languages define it as the character © though. You have to define it explicitly.
- ⇒ You cannot have the < sign inside attributes. Use < instead (defined by XML). Also it is advised to use > for the > symbol.
- ⇒ The first Comment element has two attributes named text, this is forbidden.

- ⇒ Comments <!-- --> cannot include the characters --
- ⇒ The lang attributes should be quoted.
- ⇒ XML names beginning with xml are reserved by the W3C.
Their usage should be avoided.

Here is the corrected document:

```
<?xml version="1.0" encoding="utf-16"?>
<!DOCTYPE movies [
  <!ENTITY Copy "&#169;">
]>
<movies>
  <Movie id="56225">
    <title>Love Story</title>
    <title></title>
    <year>1980</year>
    <-director name='Coppola'></director>
    <Comment text="Five start"/>
    <Comment text="Average"/>
    <newComment text="An &lt; ; important&gt;
      text">Oscar</newComment>
    <Comment lang="de">&Copy; 1980 Warner Bros.
      </comment>
    <!-- Famous movie of the 80s -->
  </Movie>
</movies>
```

18. Is this correct in HTML? How about in XHTML? Why?

```
<html>
  <head>
    <title>Untitled</title>
  </head>
  Dear Jane <br>
  <p>You are invited at the weekly meeting
  <p>Yours sincerely, <br>
  John
</html>
```

(18)

It is not well-formed XML.

The br and P tags are not closed. The following would be well-formed XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<html>
  <head>
    <title>Untitled</title>
  </head>
  <body>
    Dear Jane <br/>
    <p>You are invited at the weekly meeting
    </p>
    <p>Yours sincerely, <br/>
    John</p>
  </body>
</html>
```

But XHTML is more than just XML: It also has to have a certain structure (this is called to be "Valid"). Among others, the tags have to live in the XHTML namespace (which is a little bit like a family name), and the text in the body has to be embedded in a div tag:

```
<?xml version="1.0" encoding="UTF-8"?>
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>Untitled</title>
    </head>
    <body>
        <div>Dear Jane
            <p>you are invited at the weekly meeting</p>
            <p>Yours sincerely,<br/>
                John</p>
        </div>
    </body>
</html>
```

Valid XML documents as well as namespaces will be studied in detail later in this course.

19. Validation of XML with a DTD. Given is the following DTD:

```
<!ELEMENT movies (Movie+) >
<!ELEMENT Movie
  ( title, year, _director, (comment | newcomment)+)>
<!ATTLIST Movie id ID #REQUIRED>
<!ELEMENT title (#PCDATA)>
<!ELEMENT year (#PCDATA) >
<!ELEMENT _director (#PCDATA)>

<!ATTLIST _director name CDATA #IMPLIED>
<!ELEMENT comment (#PCDATA)>
<!ELEMENT newcomment (#PCDATA)>
<!ATTLIST comment lang CDATA #IMPLIED>
```

- a) Correct the document from Exercise Sheet 1, Exercise 2.1, to conform to this DTD.

```
<!ATTLIST Movie id ID #REQUIRED>
<!ELEMENT title (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT_director (#PCDATA)>
<!ELEMENT _director nameCDATA #IMPLIED>
<!ELEMENT Comment (#PCDATA)>
<!ELEMENT newComment (#PCDATA)>
<!ELEMENT Comment lang CDATA #IMPLIED>
<!-->
<movies>
```

```
<Movie id = "y56225">
  <title>Love Story </title>
  <year>1980</year>
  <-director name = "Coppola"/>
  <Comment lang = " " />
  <newComment>Oscar</newComment>
  <Comment lang = "de">1980 Warner Bros.</Comment>
  <!-- Famous movie of the 80s -->
</Movie>
</movies>
```

b) What is the difference between CDATA and PCDATA?

(b)

- ◇ PCDATA means "parsed character data". It means that this character data is to be parsed in particular:-
 - ⇒ Entity references (< ; > ; ' ; " ; and &) will be resolved (to <> " and & respectively), as well as any additional entities defined in the DTD.
 - ⇒ It may not contain any unencoded < or & characters, because they would be confused with an opening tag or an entity reference.
- ◇ In the DTD, PCDATA is used to say that an element may only contain parsed character data, without child elements.

- CDATA is more confusing, because it can have several meanings in the XML world:
- ⇒ In the DTD, it is used to give the most general type for an attribute:-
An attribute of type CDATA may contain any attribute value. Note, however, that entity references are resolved, and that & and < must be encoded as well. In addition, the single quote ' must be encoded to ' if the attribute value is single-quoted, and the double quote " must be encoded

to & quot; if the attribute value of double - quoted .

⇒ Where PCDATA is expected in an element, one can explicitly use a CDATA construct to escape the special XML characters like < or >, which will not be recognized as markup.

The only sequence recognized as markup in a CDATA section is]]>, which is interpreted as the end of the CDATA section.

```
<! [CDATA[  
if (a<2) { // notice the use of < without needing  
    to encode it as &lt;  
    writeln("The number is too low");  
}  
]]>
```

c) Provide four reasons for using elements instead of attributes.

Answer :-

- (c) This is big debate between programmers.
 - ⇒ Some say that attributes are for metadata whereas elements contains information
 - ⇒ In general, an element is better if one wants to nest children.
 - ⇒ In general, an element is better if there is a lot of data inside.
 - ⇒ One has to use an element if one wants to nest children.
 - ⇒ Attributes are in a set, whereas two sibling elements may have the same name.

20. Correct the mistakes in the following (not well-formed) XML document:

```
<?xml version="1.0"?>
<?DOCTYPE "eth" SYSTEM "eth.dtd"?>
<eth xmlns="http://www.ethz.ch"
      xmlns:xmlDb="http://www.dbis.ethz.ch"
      date="11.11.2006"
      xmlDb:date="12.11.2006">
  <date>13.11.2006</date>
  <president number="1">Empty</president>
  <Rektor>Name 2</rektor>
</Doc>
```

Answer :

20

⇒ eth had a wrong closing tag, so had Rektor.

⇒ a / was missing for the president tag.

⇒ The DOCTYPE should be introduced with <! and the root element should not be quoted.

```
<?xml Version="1.0"?>
<!DOCTYPE eth SYSTEM "eth.dtd">
<eth xmlns="http://www.ethz.ch"
      xmlns:xmlDb="http://www.dbis.ethz.ch"
      date="11.11.2006"
      xmlDb:date="12.11.2006">
  <date>13.11.2006</date>
  <President number="1">Empty</President>
  <Rektor>Name 2</Rektor>
</eth>
```

a) What are the namespaces of each attribute and element?

① <eth xmlns="http://www.ethz.ch"
eth is in the namespace http://www.ethz.ch
xmlns:xmlDb="http://www.dbis.ethz.ch"
date = "11.11.2006"

data does not inherit the namespace from the root element.
So it is in no namespace.

Unlike elements, an attributes with no prefix is in no
namespace, even if there is a default namespace.

xmlDb:date = "12.11.2006">

this attribute is in the http://www.dbis.ethz.ch namespace.
It is allowed to have two attributes with the same local name
if their namespaces are different.

<date>13.11.2006</date>

<President number="1">Empty</President>

the number attribute is in no namespace.

<Rektor>Name 2</Rektor>

All children elements are in the namespace of the root, i.e. http://
www.ethz.ch

</eth>

- b) Try to define the external DTD eth.dtd so that the document is valid. Does DTD actually handle namespaces?

(b) DTDs are not aware of namespaces. They see prefix bindings as normal attributes.
This means that the bindings in the documents have to be explicitly declared in the DTD (as #FIXED attribute values).

The following DTD could be used :-

```
<!ELEMENT eth (date, president, Rektor)>
<!ATTLIST eth xmlns CDATA #FIXED "http://www.
                           ethz.ch"
               xmlns:xmlDb CDATA #FIXED
                           "http://www.xmlDb.ethz.ch"
               date CDATA #IMPLIED
               xmlDb:date CDATA #IMPLIED>
<!ELEMENT date (#PCDATA)>
<!ELEMENT president (#PCDATA)>
<!ATTLIST president number CDATA #IMPLIED>
<!ELEMENT Rektor (#PCDATA)>
```