

31 Tuesday

11/20/8010

DBMS MAKEUP

26/07/20

Solution Explanation for calculating minimal canonical form.

Sol<sup>n</sup>: Step 1:-

Take input of number of elements present in domain.

Step 2:- Keep a track of all those present (possibly in map structure).

Step 3:- Take the input of functional dependency

Ex: Say if Relation is  $AB \rightarrow CD$ ,  
Take input as  $AB - CD$

Step 4 While taking input, check if all parameters are valid, if yes  $\rightarrow$  proceed  
if no  $\rightarrow$  terminate.

Step 5 While taking input perform union structure (function) on input for simplicity.

Ex: If input is  
 $A \rightarrow CD$

and already if  $A \rightarrow AB$

then store input as  $A \rightarrow BCD$

M	T	W	T	F	S	S	Wk
4	5	6	7	8	9	10	23
11	12	13	14	15	16	17	24
18	19	20	21	22	23	24	25
25	26	27	28	29	30	31	26

M	T	W	T	F	S	S	Wk
2	3	4	5	6	7	8	27
9	10	11	12	13	14	15	28
16	17	18	19	20	21	22	29
23	24	25	26	27	28	29	30

M	T	W	T	F	S	S	Wk
2	3	4	5	6	7	8	27
9	10	11	12	13	14	15	28
16	17	18	19	20	21	22	29
23	24	25	26	27	28	29	30

AUGUST 2012

Step 6: Then apply decomposition rule to 1 Wednesday  
remove any redundancy (extraneous attribute  
on Right hand - Side.)

→ After doing this either the rule is  
extraneous as whole

→ Or there is redundant (extraneous attribute)  
on Left hand Side only.

Ex:  $A \rightarrow BC$  is decomposed as  $A \rightarrow B, A \rightarrow C$ .

Step 7:

Check of extraneous rules. (fd)

for doing so,

$i \in 1, 2, \dots, n$  where  $n$  is number of rules

we have → deleted wrong [to check if rule  
is extraneous / deleted.

To check if any rule  $AB \rightarrow CD$   
needs to be deleted or not,

get enclosure using rule / Not using  
above rule

→ If both are same,  
delete the rule.

August							September						
M	6	13	20	27			3	10	17	24			
T	7	14	21	28			4	11	18	25			
W	8	15	22	29			5	12	19	26			
T	9	16	23	30			6	13	20	27			
F	10	17	24	31			7	14	21	28			
S	11	18	25				8	15	22	29			
S	12	19	26				9	16	23	30			



2 Thursday

Rakhee Handwritten

Step 88 Getting enclosure for deleting rule.

- Start with states on L.H.S.
- for including rule → unlock states on R.H.S.

for each rule make a visited array to see if it is visited.

→ if  $i$  in  $1 \dots n$ ;

vis[i] = 0

→ string set enclosure? = empty

set.push (LHS states)

set.push (R.H.S. states) →



vis[i] = true

→ Now repeat a loop

→ An loop do not visit deleted or already visited states.

→ If a rule is unvisited,

check if we have

July							August						
M	2	3	4	5	6	7	M	6	13	20	27	34	35
T							T	7	14	21	28		
W							W	8	15	22	29		
T							T	1	8	16	23	30	
F							F	2	9	17	24	31	
S							S	3	10	18	25		
Wk	26	27					Wk	4	11	19	26		
								5	12	20	27		
								6	13	21	28		
								7	14	22	29		
								8	15	23	30		
								9	16	24	31		
								10	17	25			
								11	18	26			
								12	19	27			
								13	20	28			
								14	21	29			
								15	22	30			
								16	23	31			
								17	24				
								18	25				
								19	26				
								20	27				
								21	28				
								22	29				
								23	30				
								24	31				
								25					
								26					
								27					
								28					
								29					
								30					
								31					

AUGUST 2012

3 Friday

all of its attributes on (L.H.S.) in enclosure set,

If so, then this rule can be unlocked/visited

→ visit it.

→ If new attribute is added, from R.H.S of above rule,

✓ Yes

There is a possibility that now if a rule is not visited yet,

It can be visited now → REPEAT LOOP AGAIN.

Else → If new attribute is added → TERMINATE LOOP.

→ This gives enclosure using rule

For calculating enclosure without rule all above steps remain same but  $\phi$

(do not perform it)

→ How if above two enclosures are same, ~~then~~ perform rule can be deleted. → delete it.

September							October						
1	2	3	4	5	6	7	8	9	10	11	12	13	14
15	16	17	18	19	20	21	22	23	24	25	26	27	28
29	30	31					1	2	3	4	5	6	7
							8	9	10	11	12	13	14
							15	16	17	18	19	20	21
							22	23	24	25	26	27	28
							29	30	31				



4 Saturday

Step 9: Now extra seals have been removed

Extra attributes for L.H.S. need to be deleted for each Rule: (that is not deleted).

For each part on L.H.S.

While there can be an attribute that can be deleted % of

Get the enclosure using remaining of attributes

If the rule was valid / using remaining of attributes, this rule can be

5 Sunday unlocked to its R.H.S. side,

→ Go Ahead.

Perform enclosure as per Step 8.

Now perform enclosure without this of attribute.

Now we do not have all the valid required attributes of L.H.S.,

we DO NOT INCLUDE R.H.S. here.

July							August						
M	T	W	T	F	S	Wk	M	T	W	T	F	S	Wk
2	3	4	5	6	7	8	6	7	8	9	10	11	12
9	10	11	12	13	14	15	13	14	15	16	17	18	19
16	17	18	19	20	21	22	20	21	22	23	24	25	26
23	24	25	26	27	28	29	27	28	29	30	31	32	33
30	31						30	31					35

AUGUST 2012

6 Monday

→ Perform another enclosure without the attribute that we are testing if it can be deleted.

→ In case in above process we found the attribute somewhere else in same enclosure rule.

(transitivity applies here)

→ then we can unlock / visit Rott-S of event rule.  
→ visit it.

→ Ex:  $A \rightarrow B$   
 $A \rightarrow C$

$B$  can be found from  $A \rightarrow B$ .

→ If above two enclosures are equal, this attribute is extraneous  
→ delete it.

→ Perform the same for remaining rules.

→ Finally perform the union to minimize number of rules (very imp eval test).

September							October						
M	3	4	10	17	24		M	1	8	15	22	29	
T			11	18	25		T	2	9	16	23	30	
W	5	6	12	19	26		W	3	10	17	24	31	
T			13	20	27		T	4	11	18	25		
F			14	21	28		F	5	12	19	26		
S	1	8	15	22	29		S	6	13	20	27		
Su	2	9	16	23	30		Sa	7	14	21	28		
	26		3	10	17	24			41	42	43	44	

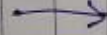
→ Finally, print the results



7 Tuesday

Flow chart

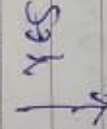
TAKE DOMAIN INPUTS



TAKE FUNCTIONAL DEPENDENCY  
INPUT



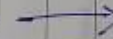
TERMINATE NO CHECK VALIDITY



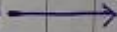
PERFORM UNION ON ALL  
FD'S



APPLY DECOMPOSITION



REMOVE INVALID RULES



CHECK FOR EXTRANEOUS ATTRIBUTES  
ON LHS IN REMAINING RULES



PERFORM UNION

PRINT RESULT

July

M	2	9
T	3	10
W	4	11
T	5	12
F	6	13
S	7	14
WL	8	15
	1	16
	2	17
	3	18
	4	19
	5	20
	6	21
	7	22
	8	23
	9	24
	10	25
	11	26
	12	27
	13	28
	14	29
	15	30
	16	31

August

6	13	20	27
7	14	21	28
8	15	22	29
9	16	23	30
10	17	24	31
11	18	25	
12	19	26	
13	20	27	
14	21	28	
15	22	29	
16	23	30	
17	24	31	
18	25		
19	26		
20	27		
21	28		
22	29		
23	30		
24	31		