

Q.1)

a)

Deterministic Finite automata

Non-deterministic finite automata

i) DFA is a finite automata which have only one transition for each input

NFA is an automata which can have multiple transition for particular input

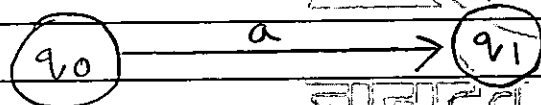
ii) DFA designing is difficult

NFA designing is not that difficult

iii) All DFAs are NFAs

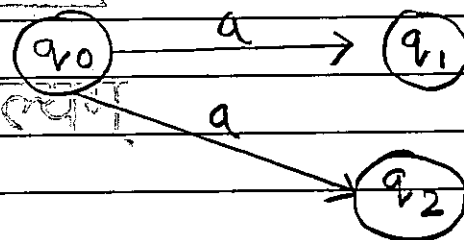
All NFAs are not DFA

iv) for ex:



ज्ञानादिव तु कैवल्यम्

for ex:



Q.1)

b)

	Q	output	1	output
$\rightarrow q_1$	q_1	1	q_2	0
q_2	q_4	1	q_4	1
q_3	q_2	0	q_3	1
q_4	q_3	0	q_2	1

In the moore machine, the output are inside the transition states,

From the given table we can observed that the q_1 and q_4 have only "1" as output and for q_2 and q_3 have "0" and "1" as output.

Therefore we can write q_2 and q_3 in the following manner:

q_2 can be split into two parts

q_{21} which have output of "1"

q_{20} which have output of "0"

Similarly

q_3 can be split into two parts

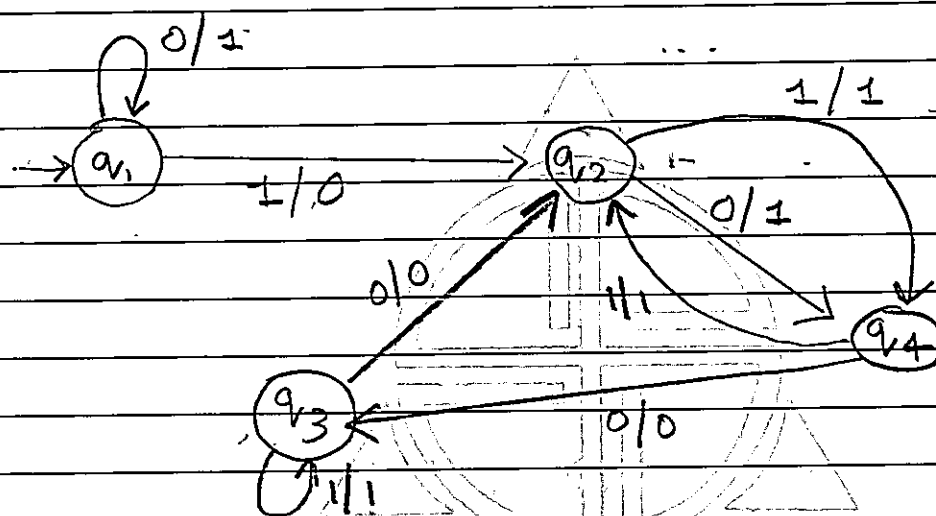
q_{31} which have output "1"

q_{30} which have output "0"

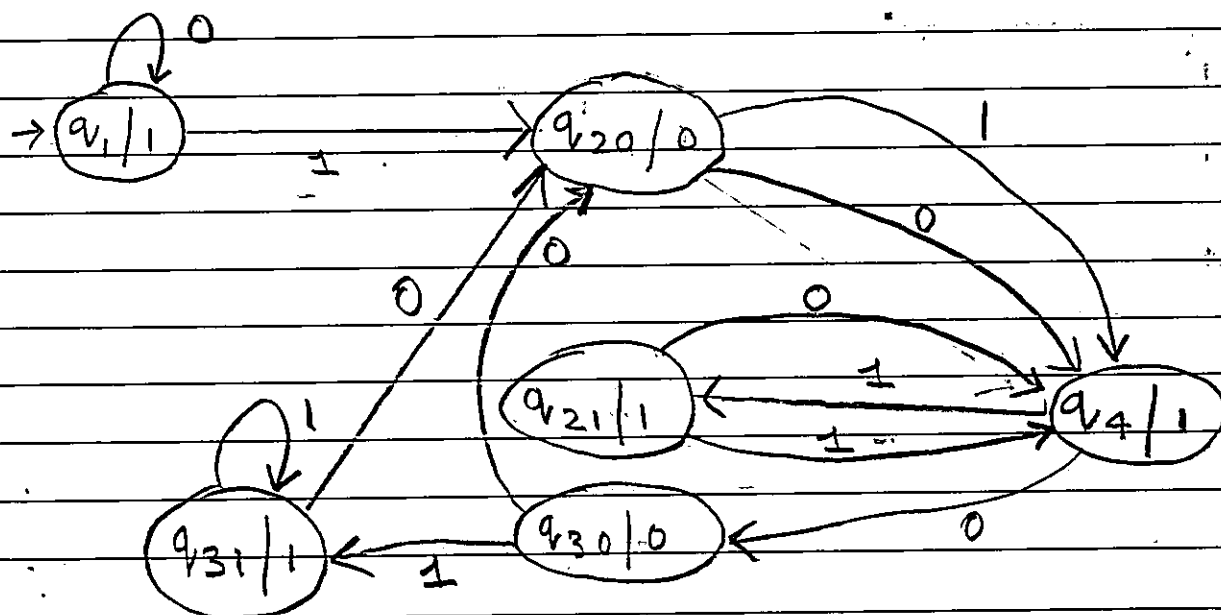
according to question, we have to Construct moore machine

\therefore Melay to Moore.

Let M be the state diagram for given melay machine.



Let M' be the moore machine:



The transition table for moore machine will be:

	0	1	output
$\rightarrow q_0$	q_1	q_{20}	1
q_{20}	q_4	q_4	0
q_{21}	q_4	q_4	1
q_{31}	q_{20}	q_{31}	1
q_{30}	q_{20}	q_{31}	0
q_4	q_{30}	q_{21}	1

← final answer

Q.1)

c)

My Hill Nerode theorem, will helps to Convert the DFA to minimize DFA.

So, first we have to make box, and by using box filling method, we have to construct a minimized DFA.

listing out the final and non-final states:

Final states: q_2, q_3, q_5, q_6

non-final states: q_1, q_4, q_7, q_8, q_0

	q_0	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8
q_0									
q_1	✓								
q_2	✓	✓							
q_3	✓	✓	✓						
q_4	✓	✓							
q_5	✓	✓	✓	✓	✓				
q_6	✓	✓	✓	✓	✓	✓			
q_7									
q_8									

We will first mark all final states...
Then, we will check one by one

$\delta(q_1, 0) = q_2$ the pairs q_2, q_1 is already
 $\delta(q_0, 0) = q_1$ marked, so we have to
 mark q_1, q_0 also.

$\delta(q_4, 0) = q_5$, the pairs q_5, q_1 is already
 $\delta(q_0, 0) = q_1$ marked so we have to mark
 q_4, q_0 .

$\delta(q_4, 0) = q_5$, q_5, q_2 is already marked
 $\delta(q_1, 0) = q_2$ therefore have to mark q_4, q_1

$\delta(q_4, 0) = q_5$ It is not marked, so we have
 $\delta(q_2, 0) = q_7$ to check for 1 input.
 $\delta(q_4, 1) = q_6$ It is unmarked pairs q_6, q_8 , so
 $\delta(q_2, 1) = q_8$ have to mark (q_4, q_2) .
 leave

$$s(q_4, 0) = q_8$$

q_5, q_8 is not marked,

$$s(q_3, 0) = q_8$$

we move for 1.

$$s(q_4, 1) = q_6$$

q_6, q_7 not marked

$$s(q_3, 1) = q_7$$

so have to leave untick

$$s(q_7, 0) = q_1$$

and $s(q_7, 1) = q_7$ so untick

$$s(q_0, 0) = q_7$$

$$s(q_0, 1) = q_4$$

$$s(q_7, 0) = q_2$$

and $s(q_7, 1) = q_7$ so untick

$$s(q_1, 0) = q_7$$

$$s(q_1, 1) = q_3$$

$$s(q_7, 0) = q_7$$

does not exist

$$s(q_2, 0) = q_7$$

$$s(q_7, 1) = q_7$$

so leave untick.

$$s(q_2, 1) = q_8$$

$$s(q_7, 0) = q_7$$

$$s(q_3, 0) = q_8$$

$$s(q_7, 1) = q_7$$

$$s(q_3, 1) = q_7$$

$$s(q_7, 0) = q_7$$

it is untick

$$s(q_4, 0) = q_5$$

$$s(q_7, 1) = q_7$$

so leave untick

$$s(q_4, 1) = q_6$$

$$s(q_7, 0) = q_7 \quad \times$$

$$s(q_5, 0) = q_7$$

$$s(q_7, 1) = q_7 \quad \text{do leave untick.}$$

$$s(q_5, 1) = q_8$$

$$s(q_7, 0) = q_7 \quad \times$$

$$s(q_6, 0) = q_7$$

$$s(q_7, 1) = q_7 \quad \times$$

$$s(q_6, 1) = q_8$$

$$s(q_7, 0) = \text{now check for } q_8.$$

$$s(q_7, 1) = q_7$$

$$s(q_8, 0) = q_8 \quad \times \quad s(q_8, 1) = q_8 \rightarrow \text{it is untick.}$$

$$s(q_0, 0) = q_1 \quad s(q_0, 1) = q_4$$

$$s(q_8, 0) = q_8 \quad \times \quad s(q_8, 1) = q_8 \quad \times$$

$$s(q_1, 0) = q_2 \quad s(q_1, 1) = q_3$$

$$s(q_8, 0) = q_8 \quad \times \quad s(q_8, 1) = q_8 \quad \times$$

$$s(q_2, 0) = q_7 \quad s(q_2, 1) = q_8$$

$$s(q_8, 0) = q_8 \quad \times \quad s(q_8, 1) = q_8 \quad \times$$

$$s(q_3, 0) = q_8 \quad s(q_3, 1) = q_7$$

$$s(q_8, 0) = q_8 \quad \times \quad s(q_8, 1) = q_8 \quad \times$$

$$s(q_4, 0) = q_5 \quad s(q_4, 1) = q_6$$

$$s(q_8, 0) = q_8 \quad \times \quad s(q_8, 1) = q_8 \quad \times$$

$$s(q_5, 0) = q_7 \quad s(q_5, 1) = q_8$$

$$s(q_8, 0) = q_8 \quad s(q_8, 1) = q_8$$

$$s(q_6, 0) = q_7 \quad s(q_6, 1) = q_8$$

$$s(q_8, 0) = q_8 \quad s(q_8, 1) = q_8$$

$$s(q_7, 0) = q_7 \quad s(q_7, 1) = q_7$$

Now again-check for untick box

$$s(q_9, 0) = q_5 \quad s(q_9, 1) = q_6$$

$$s(q_2, 0) = q_7 \quad s(q_2, 1) = q_8$$

So leave untick.

$$s(q_4, 0) = q_5 \quad s(q_4, 1) = q_6$$

$$s(q_3, 0) = q_8 \quad s(q_3, 1) = q_7$$

Now pair up the unmarked boxes

(q_1, q_2) and (q_4, q_3) तु केवलम्
The common is q_4

Therefore : (q_4, q_2, q_3)
and

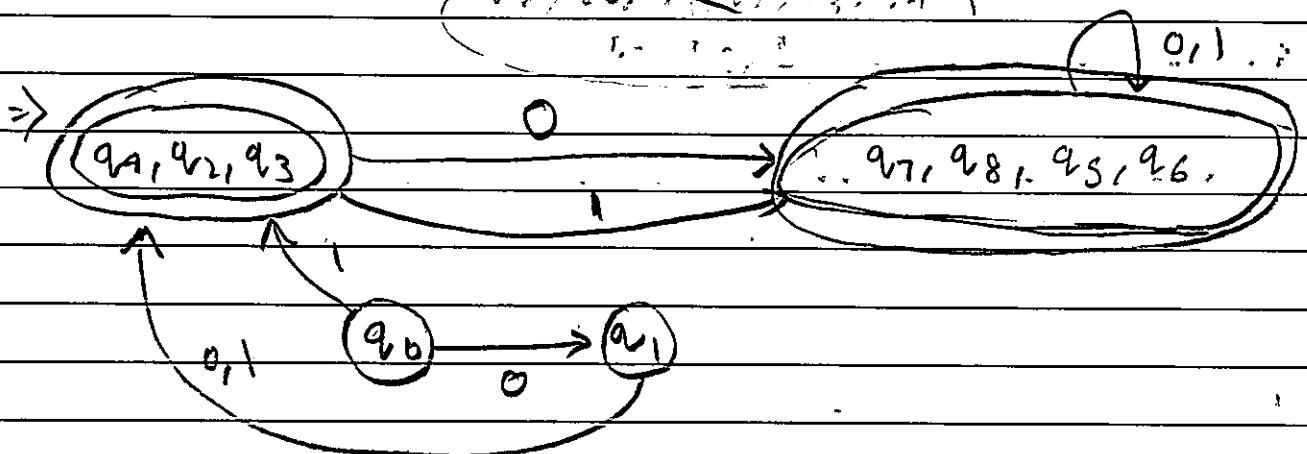
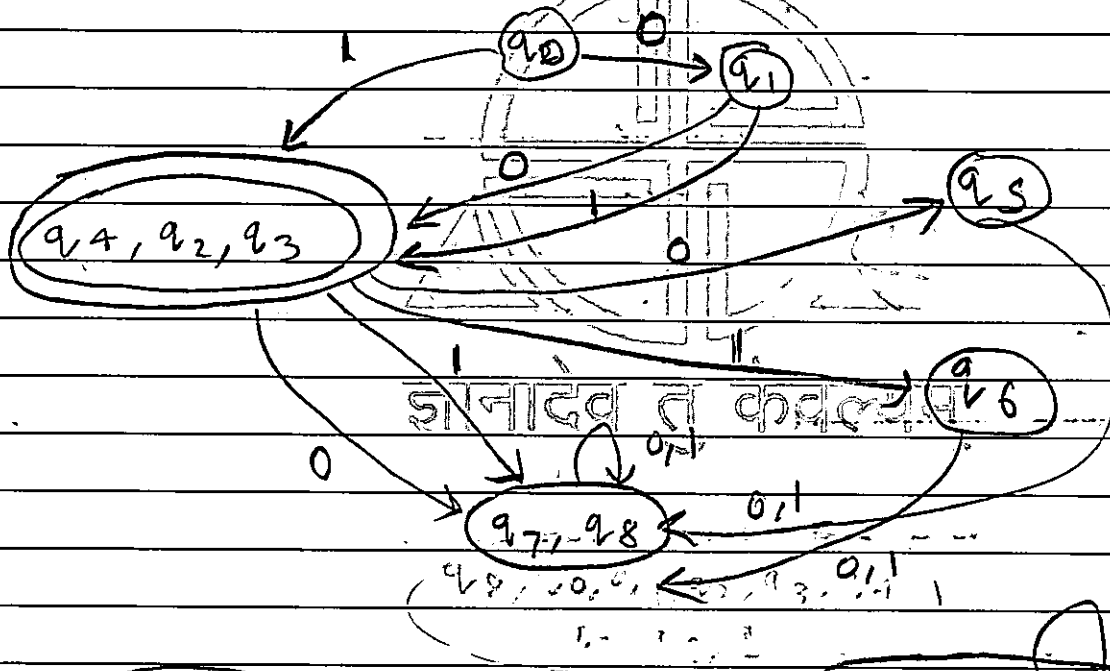
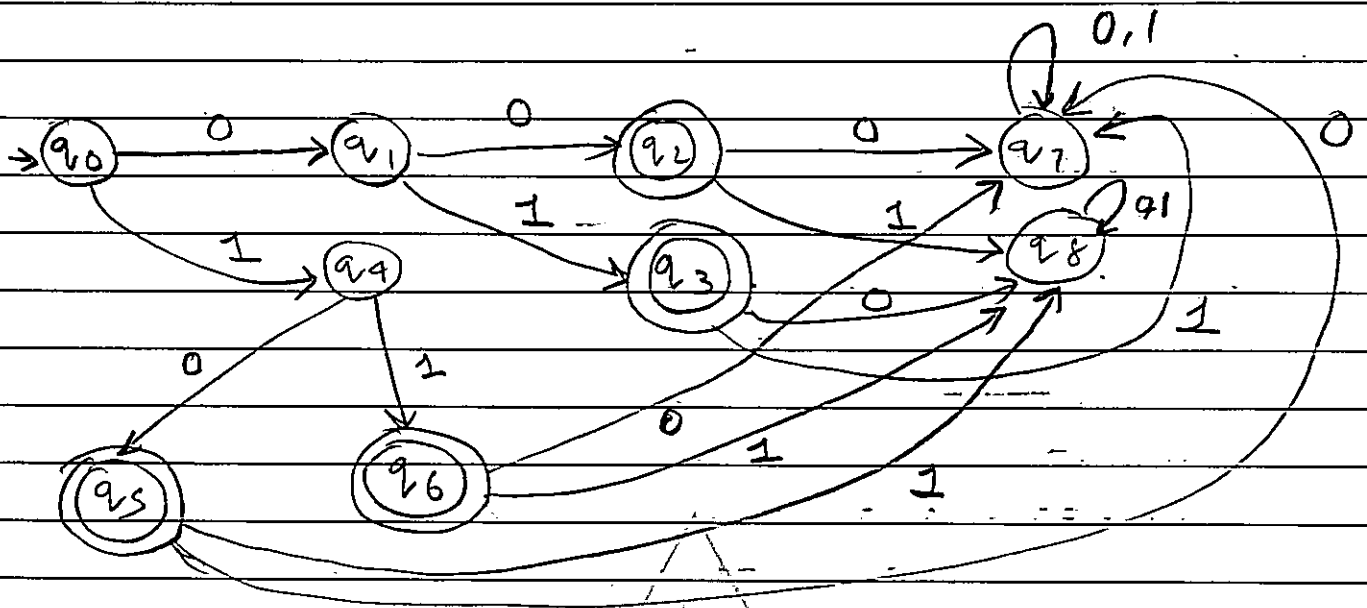
$(q_7, q_0, q_1, q_2, q_3, q_4, q_5, q_6)$

and

$(q_8, q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7)$

$q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8$

$q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8$



UNIT-2

Q.2)

a) There are various closure property of regular grammar

1) Union of regular grammar

$$L_1 = \{\epsilon, a, aa, aaa, \dots\}$$

$$L_2 = \{a, aa, aaa, \dots\}$$

$$L_1 \cup L_2 = \{\epsilon, a, aa, aaa, \dots\}$$

2) Intersection of regular grammar

$$L_1 = \{\epsilon, a, aa, aaa, \dots\}$$

$$L_2 = \{a, aa, aaa, \dots\}$$

$$L_1 \cap L_2 = \{a, aa, aaa, \dots\}$$

3) Kleen closure

$$a^* = \{\epsilon, a, aa, aaa, aaaa, \dots\}$$

4) positive closure

$$a^+ = \{a, aa, aaa, \dots\}$$

p.2)

c)

let L be a regular grammar. we have to prove that L is not regular. we will use pumping lemma theorem.

The following three conditions should be match.

i) $|y| > 0$

ii) $|xy| \leq P$

iii) for every $i \geq 0, xyz^i$, let $P = 4$, $\therefore s L = \{aaaaabbbb\}$

Case-1: $x = \epsilon, y = aaaaabbbb, z = \epsilon$

$i=2$; $(aaaaabbbb)(aaaaabbbb)$ mismatch.

Case-2: $x = a, y = aaaa, z = bbbb$

$i=2$: $aaaaaaabbbb$ mismatch.

Case-3: $x = aaaa, y = bbbb, z = b$

$i=2$; $aaaaabbbbbbb$ mismatch.

Case 4: $x = aa, y = aabb, z = bb$

$i=2$; $aaaabbaabbbb$ mismatch.

i) $|y| > 0$

$$8 > 0$$

$$4 > 0$$

$$3 > 0$$

$$4 > 0$$

ii) $|xy| \leq p$

$$8 \leq 4 \quad \times$$

$$4 \leq 4 \quad \checkmark$$

$$3 \leq 4 \quad \checkmark$$

$$4 \leq 4 \quad \checkmark$$

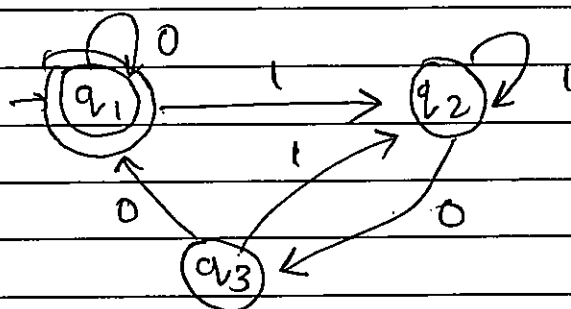
hence, from above observation
we can say that the

$L = \{a^n b^n\}$ is not regular.

निम्नलिखित में से कोई एक चुनिए

Q.2)

d)



by applying arden's theorem, we can
calculate the regular expression,

Arden's theorem, says that P and Q are transition states, then if P is not null, then,

$R = Q + RP$, it can be written as,

$$R = QP^*$$

So, first we will write the incoming states

$$q_0 = \epsilon +$$

$$q_1 = \epsilon + q_1 0 + q_3 0 \quad \text{--- (1)}$$

$$q_2 = 1q_2 + q_3 1 \quad \text{--- (2)}$$

$$q_3 = q_2 0 \quad \text{--- (3)}$$

now put eqn (3) into eqn (2)

$$\therefore q_2 = 1q_2 + (q_2 0)1$$

$$q_2 = q_2 (1 + 01) \quad \text{in the form of } R = Q + RP$$

$$q_2 = \epsilon \cdot (1 + 01)^*$$

$$\boxed{q_2 = (1 + 01)^*} \quad \text{--- (4)}$$

put the 4th eqn into q_3

$$q_3 = q_2 0$$

$$q_3 = (1+01)^* 0 \quad \text{--- (5)}$$

now, put the 5th eqⁿ into eq (1)

$$q_1 = \epsilon + q_1 0 + q_3 0$$

$$q_1 = \epsilon + q_1 0 + ((1+01)^* 0) 0$$

$$q_1 = q_1 (0 + (1+01)^* 00)$$

$$q_1 = \epsilon + \underbrace{(1+01)^* 00}_{R} + \underbrace{q_1 0}_{RP} \quad \left(R = RP^* \right)$$

$$q_1 = (1+01)^* 00 \cdot 0^*$$

so, the final state here is

q_1

$$\therefore q_1 = (1+01)^* 00 \cdot 0^*$$

final answer

Q.3)

a)

2)

all strings ending with 00

will be $(0+1)^*00$

in starting any combination of 0 and 1 can be occurred but in the end it should be 00

$\therefore (0+1)^*00$

1)

$L = \{ 00001, 000111, 000011111000, 10101, 000111100, \dots \}$

0^*

संख्या 0 का अनंत क्रम

3)

b)

chomsky classification of grammar...
is a classification of grammar as follows

Type-0 \rightarrow Unrestricted Grammar



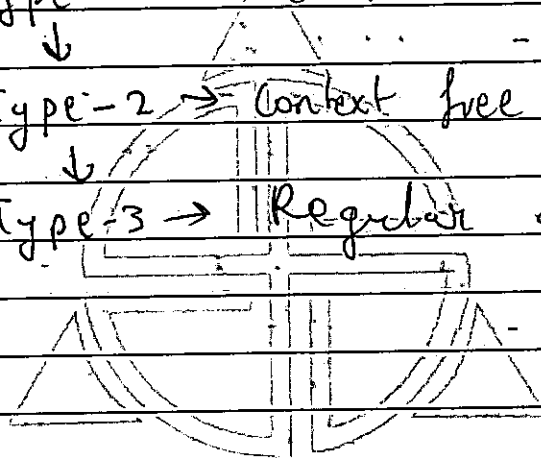
Type-1 \rightarrow Context sensitive Grammar



Type-2 \rightarrow Context free grammar



Type-3 \rightarrow Regular grammar



Type-0

It is the most powerful among all, as it is unrestricted from any kind of restriction. For ex- Turing machine, which have the ability to move left, right, in a infinite tape, with read, write head in the tape head.

It is called as unrestricted grammar because it does not have any kind of restriction, and can solve any algorithm based problem.

Type-1

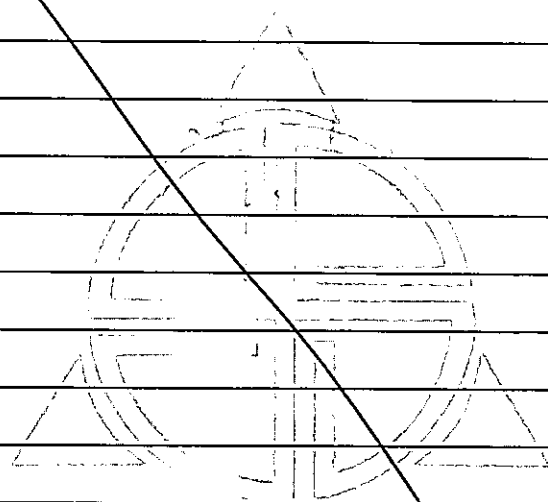
It is less powerful than Type-0, as "once we put some restrictions on Type-0, then it will ~~become~~ become Type-1. and this hierarchy is known as Context sensitive grammar and can't perform most of the tasks of Type-0.

Type-2

It is known as Context-free grammar, and once we put some restrictions on Type-1, then it is known as Type 2.

Type-3

It is known as Regular grammar, and it is the most restricted grammar of the hierarchy.



संस्कृत के अक्षर

Q.3)

d)

$S \rightarrow 0B/1A$

$A \rightarrow 0/0S/1AA$

$B \rightarrow 1S/1/0BB$

$w = 00110101$

LMD:

S

0B

00110101

0BB

0110101

1B

110101

1S

10101

0B

0101

1S

101

0B

01

X

X

RMD:

S

0B

00110101

0BB

0110101

BS

110101

B0B

110101

B1S

11101

B11A

11101

B110

1110

1110

1110

LMD:

S

0

B

0

B

B

1

B

S

0

B

S

0

B

S

19

0

B

1

RMD :

Diagram illustrating the derivation of the string "01010101" from the start symbol S using the grammar rules:

- $S \rightarrow 0B$
- $B \rightarrow 0B \mid 1S$

The derivation tree shows the step-by-step construction of the string "01010101".

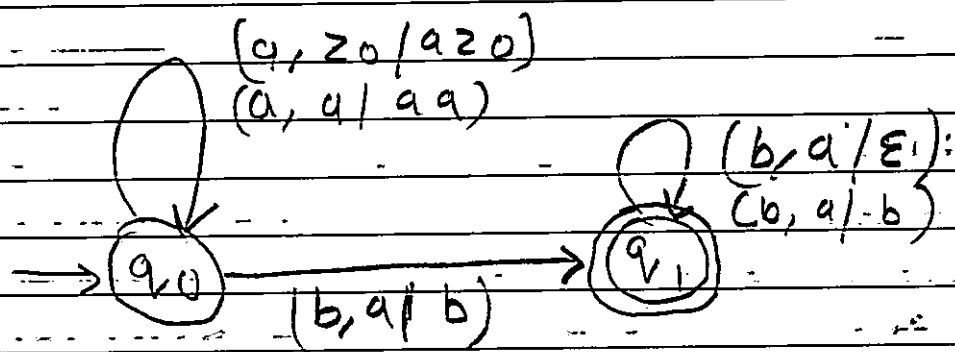
```
graph TD
    S --> 0_1[0]
    S --> B_1[B]
    B_1 --> 0_2[0]
    B_1 --> B_2[B]
    B_2 --> 1_1[1]
    B_2 --> S_1[S]
    S_1 --> 0_3[0]
    S_1 --> B_3[B]
    B_3 --> 1_2[1]
    B_3 --> S_2[S]
    S_2 --> 0_4[0]
    S_2 --> B_4[B]
    B_4 --> 1_3[1]
    B_4 --> S_3[S]
    S_3 --> 0_5[0]
    S_3 --> B_5[B]
    B_5 --> 1_4[1]
    B_5 --> S_4[S]
    S_4 --> 0_6[0]
    S_4 --> B_6[B]
    B_6 --> 1_5[1]
    B_6 --> S_5[S]
    S_5 --> 0_7[0]
    S_5 --> B_7[B]
    B_7 --> 1_6[1]
    B_7 --> S_6[S]
    S_6 --> 0_8[0]
    S_6 --> B_8[B]
    B_8 --> 1_7[1]
    B_8 --> S_7[S]
    S_7 --> 0_9[0]
    S_7 --> B_9[B]
    B_9 --> 1_8[1]
    B_9 --> S_8[S]
    S_8 --> 0_10[0]
    S_8 --> B_10[B]
    B_10 --> 1_9[1]
    B_10 --> S_9[S]
    S_9 --> 0_11[0]
    S_9 --> B_11[B]
    B_11 --> 1_10[1]
    B_11 --> S_10[S]
    S_10 --> 0_12[0]
    S_10 --> B_12[B]
    B_12 --> 1_11[1]
    B_12 --> S_11[S]
    S_11 --> 0_13[0]
    S_11 --> B_13[B]
    B_13 --> 1_12[1]
    B_13 --> S_12[S]
    S_12 --> 0_14[0]
    S_12 --> B_14[B]
    B_14 --> 1_13[1]
    B_14 --> S_13[S]
    S_13 --> 0_15[0]
    S_13 --> B_15[B]
    B_15 --> 1_14[1]
    B_15 --> S_14[S]
    S_14 --> 0_16[0]
    S_14 --> B_16[B]
    B_16 --> 1_15[1]
    B_16 --> S_15[S]
    S_15 --> 0_17[0]
    S_15 --> B_17[B]
    B_17 --> 1_16[1]
    B_17 --> S_16[S]
    S_16 --> 0_18[0]
    S_16 --> B_18[B]
    B_18 --> 1_17[1]
    B_18 --> S_17[S]
    S_17 --> 0_19[0]
    S_17 --> B_19[B]
    B_19 --> 1_18[1]
    B_19 --> S_18[S]
    S_18 --> 0_20[0]
    S_18 --> B_20[B]
    B_20 --> 1_19[1]
    B_20 --> S_19[S]
    S_19 --> 0_21[0]
    S_19 --> B_21[B]
    B_21 --> 1_20[1]
    B_21 --> S_20[S]
    S_20 --> 0_22[0]
    S_20 --> B_22[B]
    B_22 --> 1_21[1]
    B_22 --> S_21[S]
    S_21 --> 0_23[0]
    S_21 --> B_23[B]
    B_23 --> 1_22[1]
    B_23 --> S_22[S]
    S_22 --> 0_24[0]
    S_22 --> B_24[B]
    B_24 --> 1_23[1]
    B_24 --> S_23[S]
    S_23 --> 0_25[0]
    S_23 --> B_25[B]
    B_25 --> 1_24[1]
    B_25 --> S_24[S]
    S_24 --> 0_26[0]
    S_24 --> B_26[B]
    B_26 --> 1_25[1]
    B_26 --> S_25[S]
    S_25 --> 0_27[0]
    S_25 --> B_27[B]
    B_27 --> 1_26[1]
    B_27 --> S_26[S]
    S_26 --> 0_28[0]
    S_26 --> B_28[B]
    B_28 --> 1_27[1]
    B_28 --> S_27[S]
    S_27 --> 0_29[0]
    S_27 --> B_29[B]
    B_29 --> 1_28[1]
    B_29 --> S_28[S]
    S_28 --> 0_30[0]
    S_28 --> B_30[B]
    B_30 --> 1_29[1]
    B_30 --> S_29[S]
    S_29 --> 0_31[0]
    S_29 --> B_31[B]
    B_31 --> 1_30[1]
    B_31 --> S_30[S]
    S_30 --> 0_32[0]
    S_30 --> B_32[B]
    B_32 --> 1_31[1]
    B_32 --> S_31[S]
    S_31 --> 0_33[0]
    S_31 --> B_33[B]
    B_33 --> 1_32[1]
    B_33 --> S_32[S]
    S_32 --> 0_34[0]
    S_32 --> B_34[B]
    B_34 --> 1_33[1]
    B_34 --> S_33[S]
    S_33 --> 0_35[0]
    S_33 --> B_35[B]
    B_35 --> 1_34[1]
    B_35 --> S_34[S]
    S_34 --> 0_36[0]
    S_34 --> B_36[B]
    B_36 --> 1_35[1]
    B_36 --> S_35[S]
    S_35 --> 0_37[0]
    S_35 --> B_37[B]
    B_37 --> 1_36[1]
    B_37 --> S_36[S]
    S_36 --> 0_38[0]
    S_36 --> B_38[B]
    B_38 --> 1_37[1]
    B_38 --> S_37[S]
    S_37 --> 0_39[0]
    S_37 --> B_39[B]
    B_39 --> 1_38[1]
    B_39 --> S_38[S]
    S_38 --> 0_40[0]
    S_38 --> B_40[B]
    B_40 --> 1_39[1]
    B_40 --> S_39[S]
    S_39 --> 0_41[0]
    S_39 --> B_41[B]
    B_41 --> 1_40[1]
    B_41 --> S_40[S]
    S_40 --> 0_42[0]
    S_40 --> B_42[B]
    B_42 --> 1_41[1]
    B_42 --> S_41[S]
    S_41 --> 0_43[0]
    S_41 --> B_43[B]
    B_43 --> 1_42[1]
    B_43 --> S_42[S]
    S_42 --> 0_44[0]
    S_42 --> B_44[B]
    B_44 --> 1_43[1]
    B_44 --> S_43[S]
    S_43 --> 0_45[0]
    S_43 --> B_45[B]
    B_45 --> 1_44[1]
    B_45 --> S_44[S]
    S_44 --> 0_46[0]
    S_44 --> B_46[B]
    B_46 --> 1_45[1]
    B_46 --> S_45[S]
    S_45 --> 0_47[0]
    S_45 --> B_47[B]
    B_47 --> 1_46[1]
    B_47 --> S_46[S]
    S_46 --> 0_48[0]
    S_46 --> B_48[B]
    B_48 --> 1_47[1]
    B_48 --> S_47[S]
    S_47 --> 0_49[0]
    S_47 --> B_49[B]
    B_49 --> 1_48[1]
    B_49 --> S_48[S]
    S_48 --> 0_50[0]
    S_48 --> B_50[B]
    B_50 --> 1_49[1]
    B_50 --> S_49[S]
    S_49 --> 0_51[0]
    S_49 --> B_51[B]
    B_51 --> 1_50[1]
    B_51 --> S_50[S]
    S_50 --> 0_52[0]
    S_50 --> B_52[B]
    B_52 --> 1_51[1]
    B_52 --> S_51[S]
    S_51 --> 0_53[0]
    S_51 --> B_53[B]
    B_53 --> 1_52[1]
    B_53 --> S_52[S]
    S_52 --> 0_54[0]
    S_52 --> B_54[B]
    B_54 --> 1_53[1]
    B_54 --> S_53[S]
    S_53 --> 0_55[0]
    S_53 --> B_55[B]
    B_55 --> 1_54[1]
    B_55 --> S_54[S]
    S_54 --> 0_56[0]
    S_54 --> B_56[B]
    B_56 --> 1_55[1]
    B_56 --> S_55[S]
    S_55 --> 0_57[0]
    S_55 --> B_57[B]
    B_57 --> 1_56[1]
    B_57 --> S_56[S]
    S_56 --> 0_58[0]
    S_56 --> B_58[B]
    B_58 --> 1_57[1]
    B_58 --> S_57[S]
    S_57 --> 0_59[0]
    S_57 --> B_59[B]
    B_59 --> 1_58[1]
    B_59 --> S_58[S]
    S_58 --> 0_60[0]
    S_58 --> B_60[B]
    B_60 --> 1_59[1]
    B_60 --> S_59[S]
    S_59 --> 0_61[0]
    S_59 --
```


Q. 4)

b)

$$L = \{ a^n b^{2n} \mid n \geq 1 \}$$

$$L = \{ a^4 b^5, a^3 b^6, a^2 a b^5 b^5, b^6 \}$$



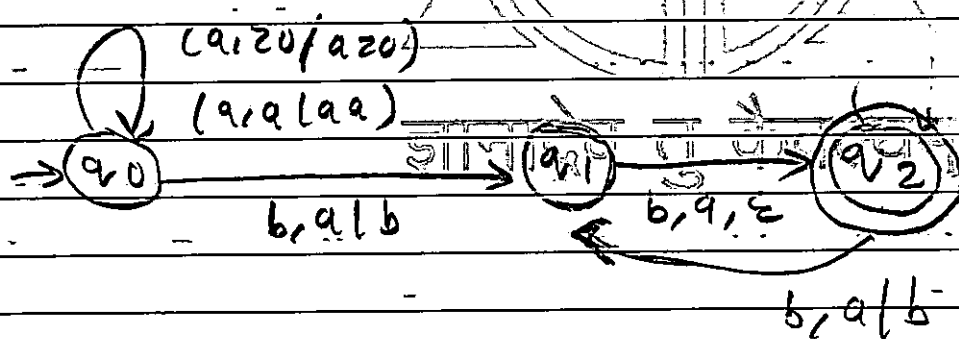
$$\delta(q_0, a, z_0) = (q_0, az_0)$$

$$\delta(q_0, a, a) = (q_0, aa)$$

$$\delta(q_0, b, a) = (q_1, b)$$

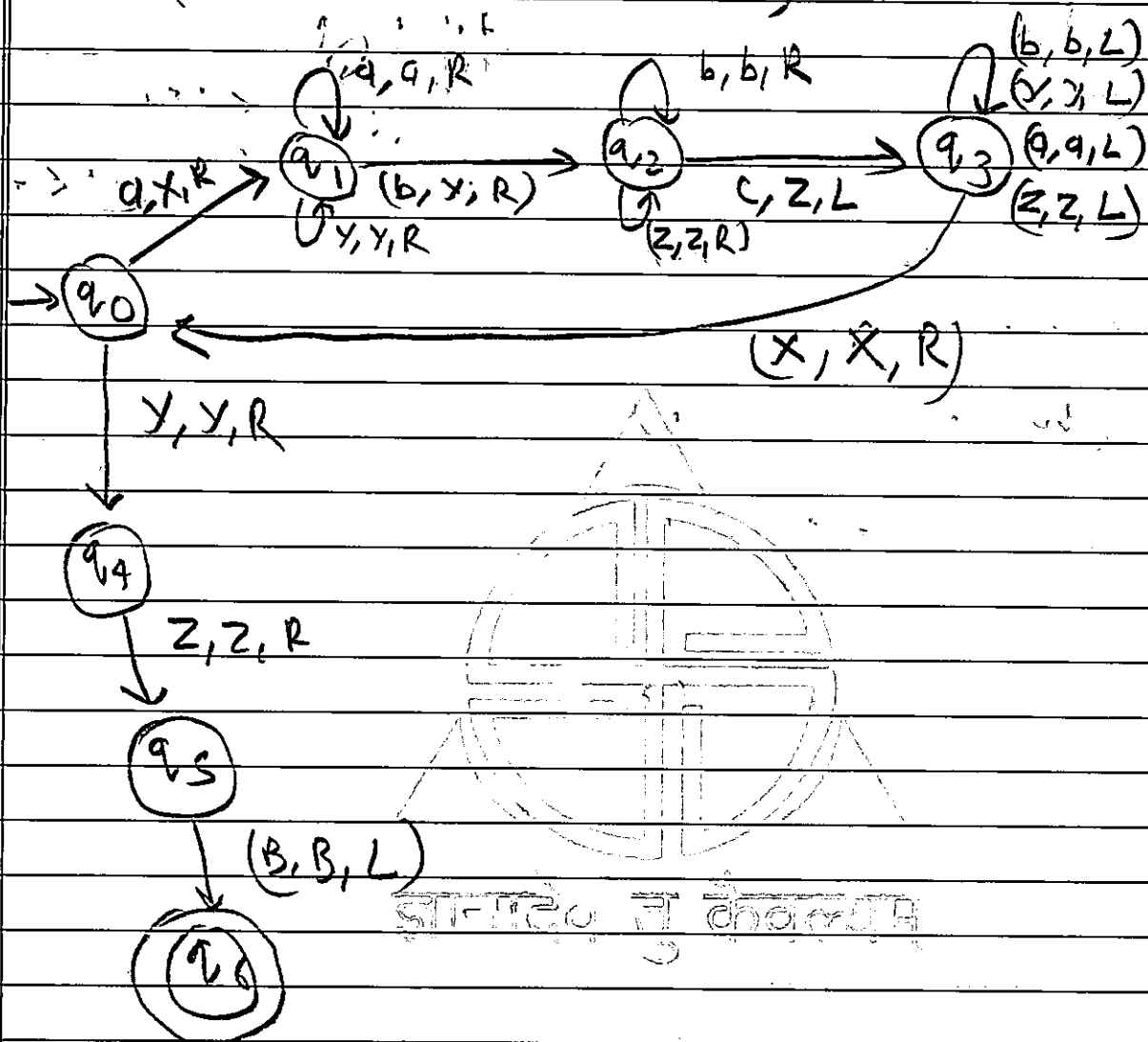
$$\delta(q_1, b, a) = (q_1, \epsilon)$$

$$\delta(q_1, b, a) = (q_1, b)$$



Q.4)

c) $L = \{ BBBaaabbbCCCBB \}$



Q.5)

a)

partial function is the function which determines the not every solution, for a particular problem.
for ex:

$$f(n) = \frac{1}{a} ; a \neq 0 \text{ (it does}$$

not define for $a=0$).

initial function can be of three types

i) zero function $z(x) = 0$.

ii) successor function $s(x) = x + 1$

iii) Projection function $U_2^3(0, 1, 2) = 1$

Q.5)

b) $f(n, y) = n * y$

To prove, it is primitive we have to follow

initial, composition, recursive function

$$f(x, y) = xy$$

$$i) \text{ put } y = 0$$

$$f(x, 0) = 0$$

$$z(f(x, 0)) = 0$$

$$ii) \text{ put } y = y + 1$$

$$f(x, y) = xy$$

$$= x(y + 1)$$

$$= xy + 1$$

$$= S(f(x, y)) = xy + 1$$

\therefore

ज्ञानादेव त केवलमस्य

$$g(x, y, f(x, y)) = \cup_3 (x, y, S(f(x, y)))$$

$$\Rightarrow f(x, y) = xy$$

$$i) \text{ put } y = 0$$

$$f(x, 0) = x^0 = 1$$

$$\therefore S(z(f(x, y))) = S(0) = 1,$$

ii) put $y = y + 1$

$$x^y = x^{y+1}$$

$$= \underline{x^y + x^y}$$

$\cdot f(n, y) \neq$ when $y \geq 0$, $\therefore x$.

$$\therefore U_3^2(x, z(y), f(n, y)) + U_3^3(x, y, f(n, y))$$

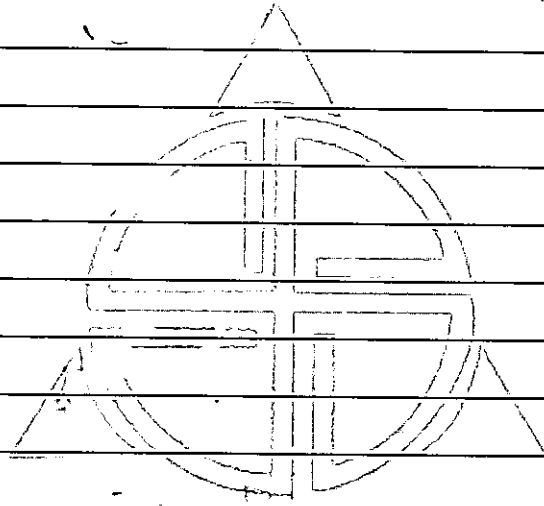
Q-5)

c)

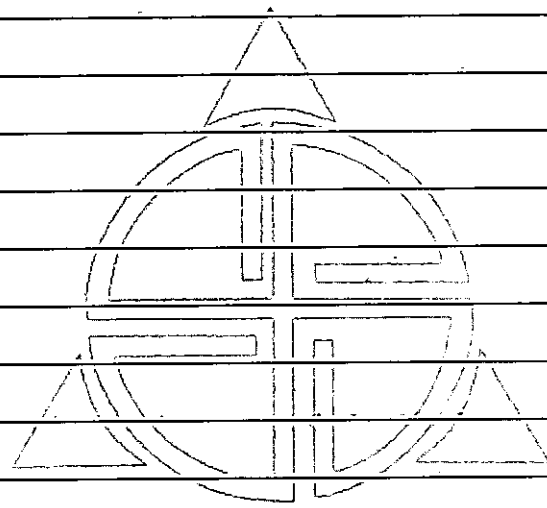
space time complexity $O(n)$

time complexity depends on length of
 Turing machine \therefore

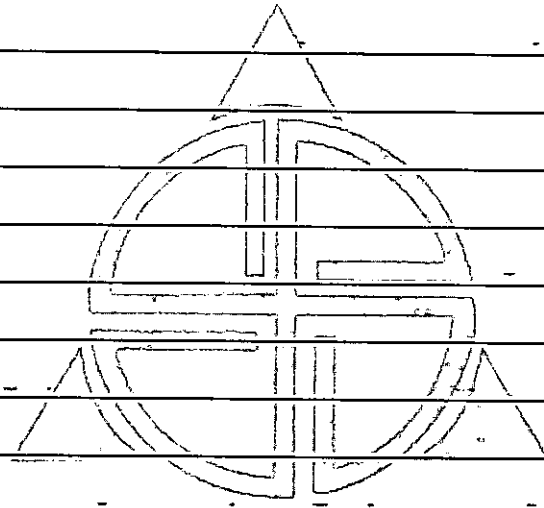
$$O(n \log n)$$



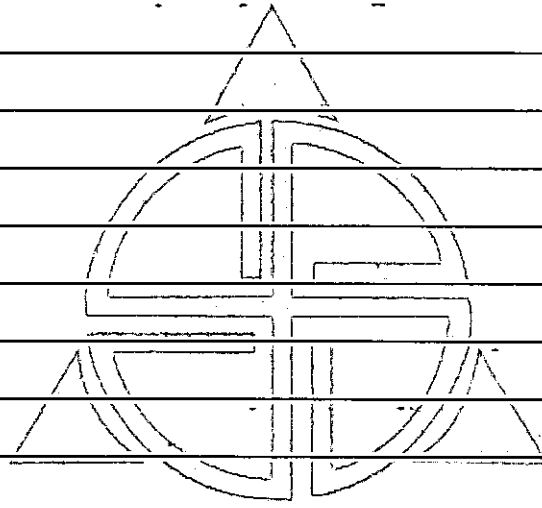
॥ ज्ञानं तं केवलम् ॥



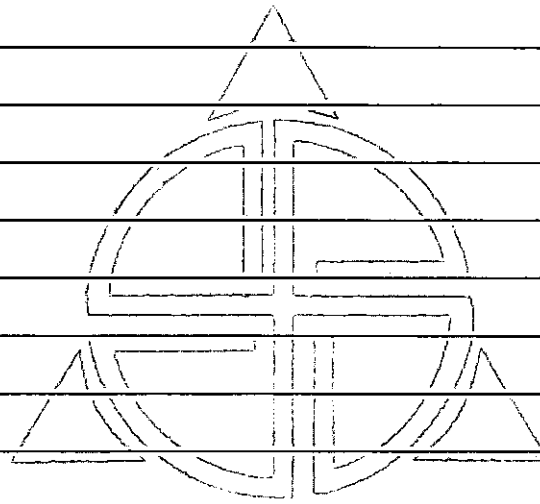
ज्ञानादेव तु कैवल्यम्



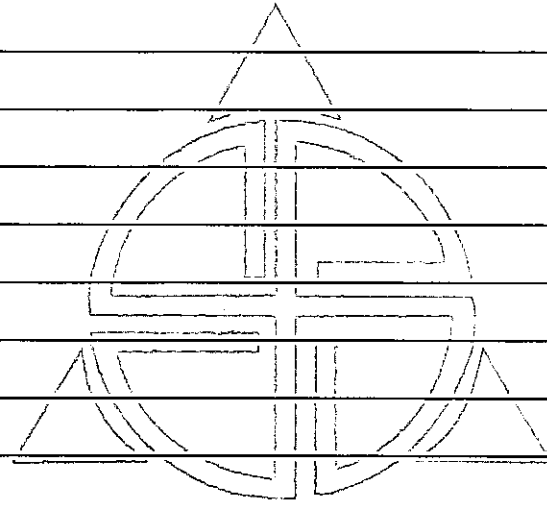
ज्ञानादेव तु कैवल्यम्



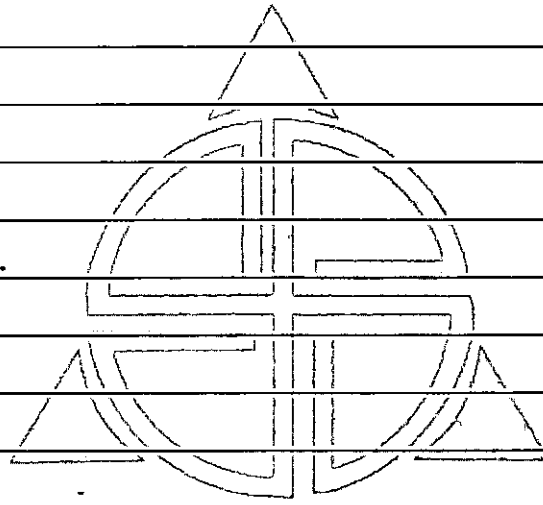
ज्ञानादेतं तु कैवल्यम्



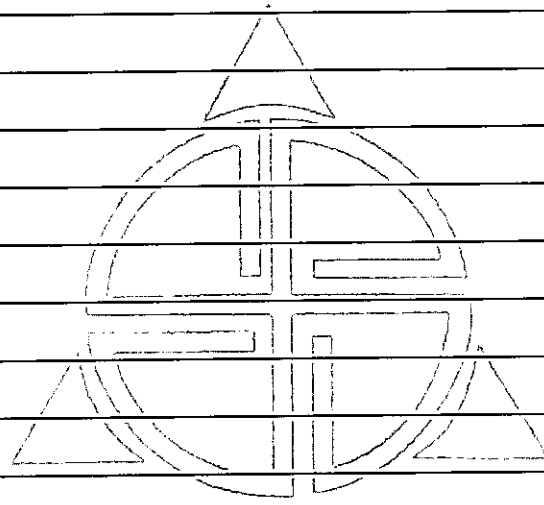
ज्ञानादेव तु कैवल्यम्



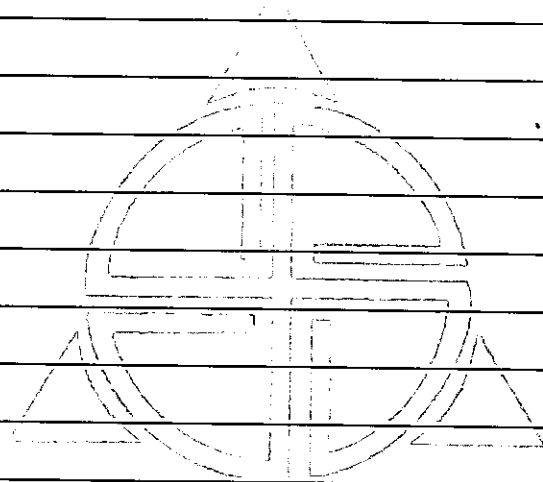
ज्ञानादेव तु कैवल्यम्



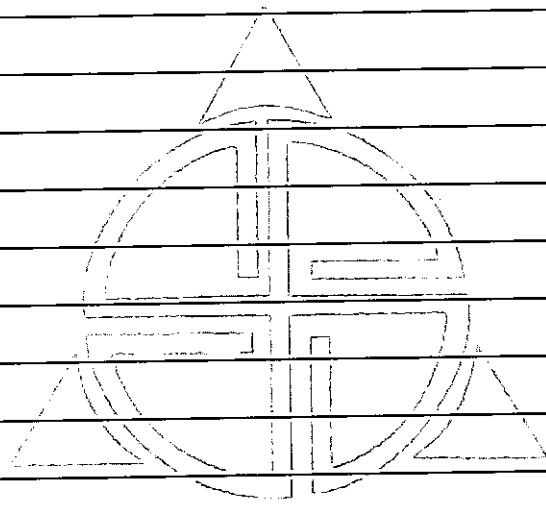
ज्ञानादेव तु कैवल्यम्



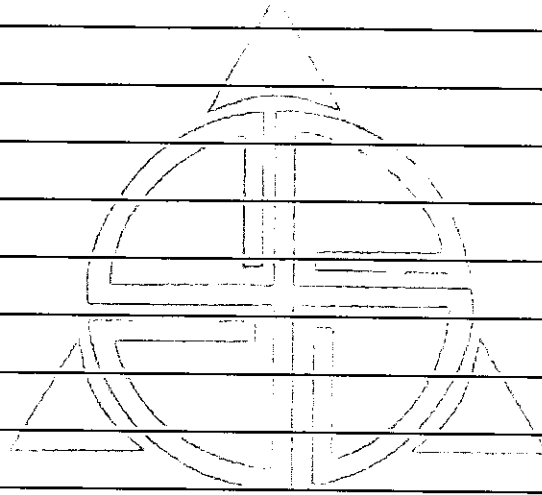
ज्ञानादेव तु कैवल्यम्



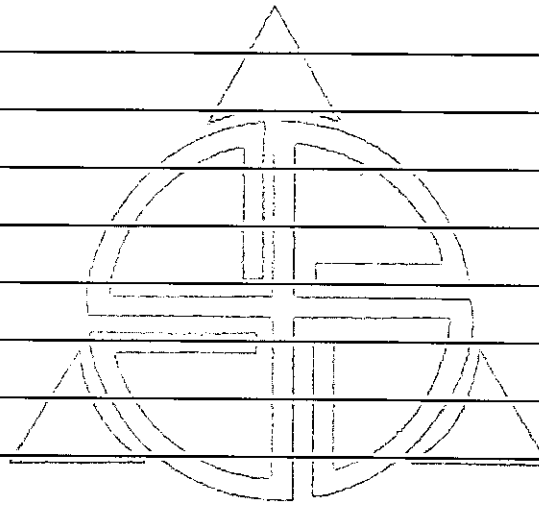
संन्यासः न संन्यासः



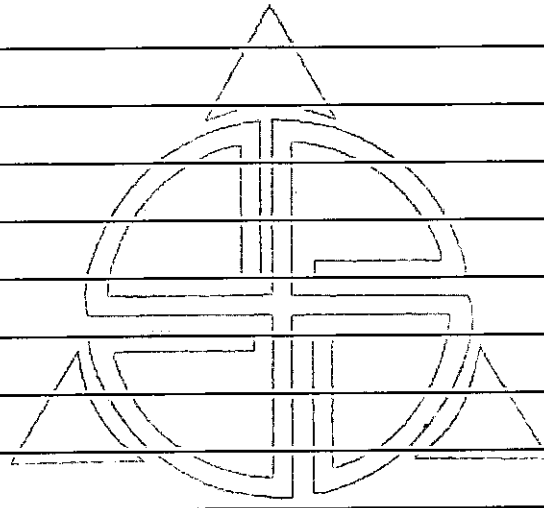
ज्ञानादेव तु कैवल्यम्



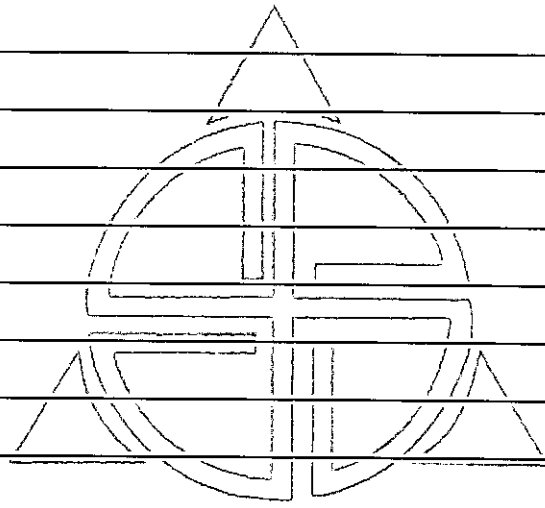
सोमादेव तु वैश्वदेवम्



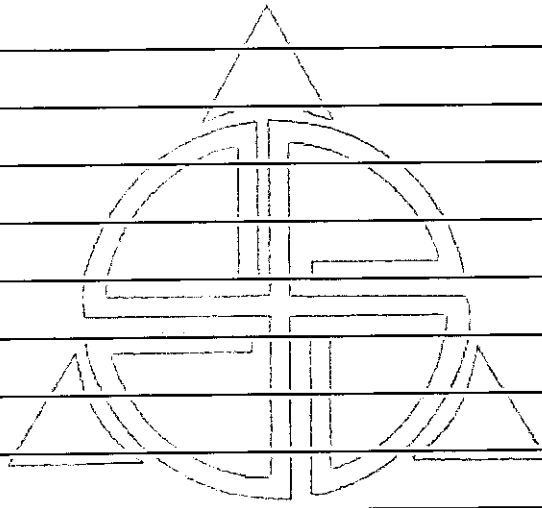
ज्ञानादेव तु कैवल्यम्



ज्ञानादेव तु कैवल्यम्



ज्ञानादेव तु कैवल्यम्



ज्ञानादेन तु कैवल्यम्

