Total No. of Questions: 6 *Total No. of Printed Pages:3*

Enrollment No.....

Faculty of Engineering

End Sem (Odd) Examination Dec-2017

CA5CO13 Theory of Computation

Programme: MCA Branch/Specialisation: Computer Application

Duration: 3 Hrs. **Maximum Marks: 60**

Note: All questions are compulsory. Internal choices, if any, are indicated. Answers of Q.1 (MCQs) should be written in full instead of only a, b, c or d.

Q.1 i. Which of the following is true? **1**

(a) (01)*0 = 0(10)*

(b) (0+1)*0(0+1)*1(0+1) = (0+1)*01(0+1)*

(c) (0+1)*01(0+1)*+1*0* = (0+1)*

(d) All of these

ii. A language is regular if and only if **1**

(a) Accepted by DFA (b) Accepted by PDA

(c) Accepted by LBA (d) Accepted by Turing machine

iii. Which of the following does not obey pumping lemma for context free languages? **1**

(a) Finite languages (b) Context free languages

(c) Unrestricted languages (d) None of these

iv. Every grammar in Chomsky Normal Form is: **1**

(a) Regular (b) Context sensitive

(c) Context free (d) All of these

v. The transition in Push down automaton makes is additionally dependent upon the: **1**

(a) Stack (b) Input tape (c) Terminals (d) None of these

vi. PDA is more powerful than **1**

(a) Turing machine (b) Finite automata

(c) Both (a) and (b) (d) None of these

vii. A turing machine that is able to simulate other turing machines: **1**

(a) Nested Turing machines

(b) Universal Turing machine

(c) Counter machine

(d) None of these

P.T.O.

viii. The value of n if turing machine is defined using n-tuples: **1**
(a) 6 (b) 7 (c) 8 (d) 5

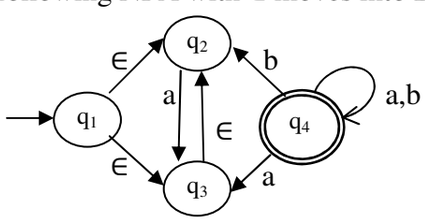
ix. A language L is said to be _____ if there is a turing machine M such that L(M)=L and M halts at every point. **1**
(a) Turing acceptable (b) Decidable
(c) Undecidable (d) None of these

x. Decidable can be taken as a synonym to: **1**
(a) Recursive (b) Non recursive
(c) Recognizable (d) None of these

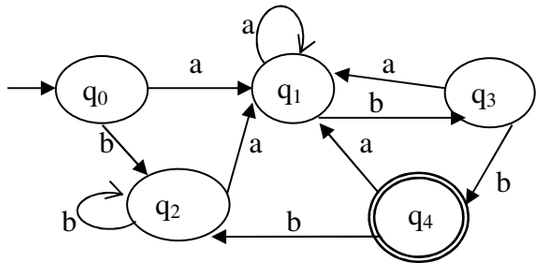
Q.2 i. What do you mean by principle of induction? **2**

ii. Briefly explain the difference between DFA and NDFA. **3**

iii. Convert the following NFA with ∈ moves into DFA. **5**



OR iv. Construct a minimum state automaton for the following DFA. **5**



Q.3 i. Define the following **3**
(a) Chomsky Normal form (b) Greibach Normal form

ii. Convert the given CFG to CNF **7**
S⟶aSa | bSb | a | b

OR iii. Remove all unit production, all useless productions and all ∈ *productions* from the grammar. **7**
S ⟶ aA | aBB
A⟶ aaA | ∈
B⟶ bB | bbC And C ⟶B

Q.4 i. What are the different types of languages accepted by a pushdown Automata and define them? **3**

ii. Construct a PDA for the following set: **7**
L = { $a^{2n} b^n$ | n>=1}

OR iii. Convert the following context free grammar into PDA : **7**
S⟶ aS | aA
A⟶ bA | b
Take the input as aab:

Q.5 i. What is Turing Machine? Explain the configuration of Turing machine. **4**

ii. Construct a Turing machine for L = { $a^n b^n c^n$ | n>=1}. **6**

OR iii. Construct a Turing machine for the language of even number of 1's and even number of 0's over ∑ = {0,1}. **6**

Q.6 Attempt any two:

i. What do you mean by 'Decidable' and 'Undecidable' problems? Give example. **5**

ii. Explain P And NP Class problem in detail with example. **5**

iii. Write short note on the following:
(a) Recursive enumerable set (b) Travelling Salesmen problem **5**

******

NAME
ENROLMENT No.
SUBJECT

End sem (Odd) Examinations

CA5C013 - TOC.

Answer:

Qu-1 i) All the mentioned.

ii) Accepted by DFA.

iii) Unrestricted language.

iv) Context Free.

(v) stack.

(vi) ~~Turing M/C~~ Finite automata

(vii) Universal turing m/c.

(viii) 7

(ix) Decidable

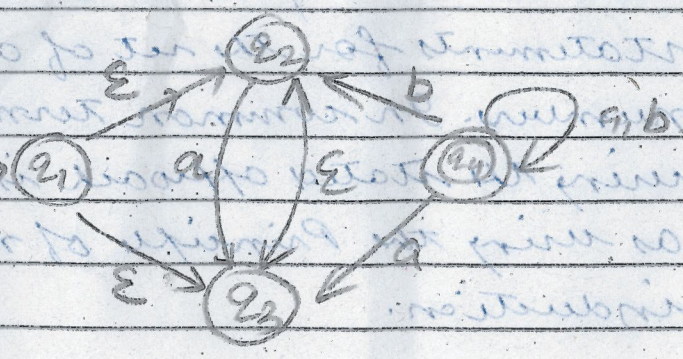
(x) Recursive

Qu-2 i) Mathematical Induction is a mathematical proof technique used to prove a given statement about all-well ordered set. Most commonly it is used to establish statements for the set of all natural numbers. In common terminology, using the stated approach is referred to as using the Principle of mathematical induction.

ii) The difference between DFA & NDFA are

	DFA	NFA
(i)	For every symbol of the Alphabet, there is only one state Transition in DFA.	We do not need to specify how does the NFA react according to the symbol.
ii)	DFA cannot use empty string transition.	NFA can use empty string transition.
iii)	DFA can be understood as one machine.	NFA can be understood as multiple.
iv)	Backtracking is allowed in DFA.	Backtracking is not allowed in NFA
v)	DFA is more difficult to construct.	NFA is easier to

iii)



ε-closure $(q_1) = \{q_1, q_2, q_3\}$ as A.

ε-closure $(q_2) = \{q_2\}$

ε-closure $(q_3) = \{q_2 q_3\}$

ε-closure $(q_4) = \{q_4\}$.

$\delta'(A, a) = \varepsilon\text{-}C(\delta(A, a))$.

$= \varepsilon\text{-}C(\delta(q_1, q_2 q_3), a)$.

$= \varepsilon\text{-}C(\delta(q_1, a) \cup \delta(q_2, a) \cup \delta(q_3, a))$

$= \varepsilon\text{-}C(\phi \cup q_3 \cup \phi)$.

$= \varepsilon\text{-}C(q_3)$.

$= \{q_2 q_3\}$: call it as B.

$\delta'(A, b) = \varepsilon\text{-}C(\delta(A, b))$.

$= \varepsilon\text{-}C(\delta(q_1, q_2 q_3), b)$.

$= \varepsilon\text{-}C(\delta(q_1, b) \cup \delta(q_2, b) \cup \delta(q_3, b))$

$= \varepsilon\text{-}C(\phi \cup \phi \cup \phi)$.

$= \phi$.

$\delta'(B, a) = \varepsilon\text{-}C(\delta(B, a))$.

$= \varepsilon\text{-}C(\delta(q_2 q_3), a))$.

$= \varepsilon\text{-}C(\delta(q_2, a) \cup \delta(q_3, a))$.

$= \varepsilon\text{-}C(q_3 \cup \phi)$.

$= \varepsilon\text{-}C(q_3)$.

$= \{q_2, q_3\}$

$\delta'(B, b) = \varepsilon\text{-}C(\delta(B, b))$.

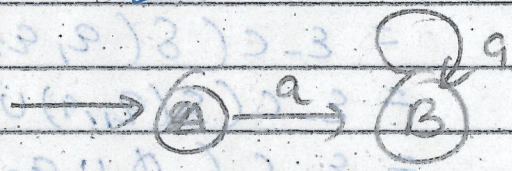
$= \varepsilon\text{-}C(\delta(q_2 q_3), b)$.

$= \varepsilon\text{-}C(\delta(q_2, b) \cup \delta(q_3, B))$.

$= \varepsilon\text{-}C(\phi \cup \phi)$

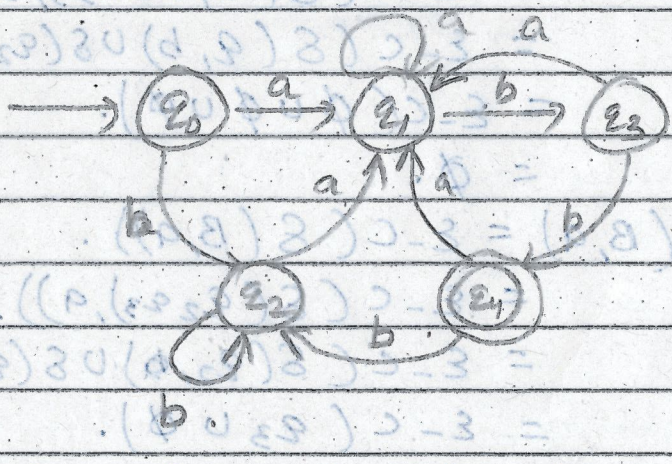
$= \phi$.

There is no final state because $A = \{q_1, q_2 q_3\}$ and $B = \{q_2 q_3\}$ and actual final state is $\{q_4\}$. Neither A or B contains q_4. Hence A and B cannot be final state required DFA will be



Thus this language accepts ϕ.

(iv) Minimum state automata.



It will be easier if we construct a transition table for the finite automata.

STATE / I/O	a	b
→ q_0	q_1	q_2
q_1	q_1	q_3
q_2	q_1	q_2
q_3	q_1	q_4
(q_4)	q_1	q_2

Now we will first obtain Q_1^0

$\therefore Q_1^0 = F = \{q_4\}$ → Final state

$Q_2^0 = Q - Q_1^0$

$= \{q_0 q_1 q_2 q_3\}$

Hence $\pi_0 = \{\{q_4\}, \{q_0, q_1 q_2 q_3\}\}$.

As we can not partitioned $Q_1^0 = \{q_4\}$ further $Q_1' = \{q_4\}$.

Now considered subset $\{q_0 q_1 q_2 q_3\}$ we will compare q_0 with q_1, q_2 & q_3.

$Q_{I/O}$	a	b
q_0	q_1	(q_2)
q_1	q_1	(q_3)

Both are 1-equ.

$Q_{I/O}$	a	b
q_0	q_1	(q_2)
q_2	q_1	(q_2)

$q_0 q_2$ are equivalent

δ	a	b
q_0	q_1	q_2
q_3	q_1	q_4

$q_2 \in Q_2^0$ & $q_4 \in Q_1^0$.

Hence both are not 1-equ.

$\therefore Q_2' = \{q_0 q_1 q_2\}$

$Q_3' = \{q_3\}$.

Hence $\pi_1 = \{\{q_4\}, \{q_0 q_1 q_2\}, \{q_3\}\}$.

Now we will compare q_0 with $q_1 q_2$

δ	a	b
q_0	q_1	q_2
q_1	q_1	q_3

δ	a	b
q_1	q_1	q_2
q_2	q_1	q_2

As $q_2 \in Q_2'$ & $q_3 \in Q_3'$ clearly both are equivalent.

Both are not 2-equ.

Hence we get

$\pi_2 = \{\{q_4\}, \{q_0, q_2\}, \{q_1\}, \{q_3\}\}$

We can now state $\{q_0 q_2\}$ are 3-equivalent and we cannot partition $\{q_2\}$ & $\{q_3\}$ further.

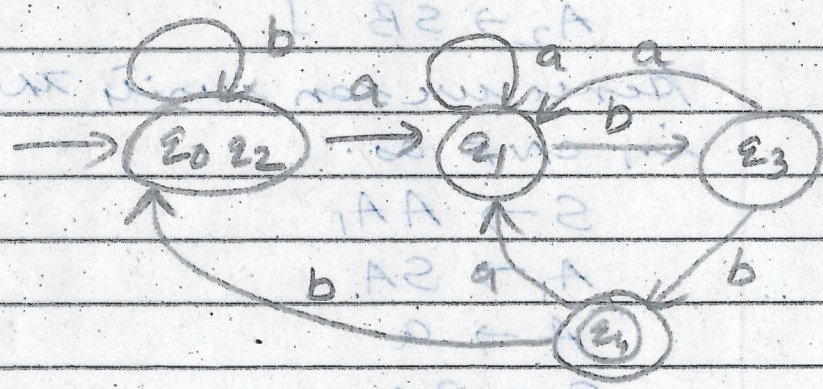
Hence.

$\pi_3 = \{\{q_4\}, \{q_0 q_2\}, \{q_1\}, \{q_3\}\}$.

As $\pi_3 = \pi_2$ and π_2 gives equivalent classes we can construct MSA as

S	a	b
$\{q_0 q_2\}$	q_1	$\{q_0 q_2\}$
q_1	q_1	q_3
q_3	q_1	q_4
q_4	q_1	$\{q_0 q_2\}$

TD



Qu-3 i)

ii) convert the given CFG to CNF

$S \to aSa / bSb / a / b.$

Let us start adding new symbol for the terminals.

$S \to ASA$.

$A \to a$

$S \to BSB$

$B \to b$.

Let us take

$S \to ASA$ for converting to CNF.

$S \to A\boxed{SA}$

Replace it by A_2

$S \to AA_1$ } Both are in CNF.

$A_1 \to SA$ }

Take $S \to B\boxed{SB}$

Replace it by A_2

$S \to BA_2$ } Both are in CNF.

$A_2 \to SB$ }

Hence we can write the Rule in CNF as.

$S \to AA_1$

$A_1 \to SA$

$A \to a$

$S \to BA_2$

$A_2 \to SB$

$B \to b$

$S \to a$

$S \to b$.

iii) Consider

$S \rightarrow aA / aBB$

$A \rightarrow aaA / \epsilon$

$B \rightarrow bB / bbC$

$C \rightarrow B$

The unit production is $C \rightarrow B$, to remove this production we will replace C by B and then eliminate $C \rightarrow B$ production.

$S \rightarrow aA / aBB$

$A \rightarrow aaA / \epsilon$

$B \rightarrow bB / bbB.$

Now consider the production rule

$B \rightarrow bB / bbB$

This will generate the string $bB, bbB, bbbB, bbbbB \ldots$ This will never terminate. Here $B \rightarrow bB / bbB$ is supposed to be an useless production. Hence we will eliminate the production containing B. Therefore the production rule becomes

$S \rightarrow aA$

$A \rightarrow aaA / \epsilon$

Now to eliminate ϵ production we will replace A by ϵ and add.

those rule in existing production rules. Here:

$$S \rightarrow aA/a$$
$$A \rightarrow aaA/aa.$$

Thus we have a reduced grammar. The given grammar represents the language containing any number of a's. There must be atleast single 'a' present in the language.

Qu-4 i)

ii) Construct a PDA

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

For constructing need a PDA we will apply a simple logic. That is, when we read a initially, push all the a's onto a stack. When we read a single b, pop two a's from the stack. This process will be repeated. Finally we read Δ the stack should be empty. The ID can be given below.

$\delta(q_0, a, z_0) = \{(q_1, a z_0)\}$

$\delta(q_0, a, a) = \{(q_1, aa)\}$

$\delta(q_0, a, a) = \{(q_0, a)\}$

$\delta(q_0, b, a) = \{(q_0, \varepsilon)\}$

$\delta(q_0, \varepsilon, z_0) = \{(q_0, \varepsilon)\}$.

consider the string aab

$\delta(q_0, aab, z_0) \vdash (q_1, ab, az_0)$
$\vdash (q_0, b, az_0)$
$\vdash (q_0, \varepsilon, z_0)$.
$\vdash (q_0, \varepsilon)$.
$\vdash$ accept.

iii) convert the following CFG to PDA

$S \to aS / aA$

$A \to bA / b$.

The PDA

$M = \{q_0\}, (a,b), (S, A, a, b), \delta, q_0, S, \phi)$.

The mapping function δ can be

$\delta(q_0, \varepsilon, S) = \{(q_0, aS), (q_0, aA)\}$

$\delta(q_0, \varepsilon, A) = \{(q_0, bA), (q_0, b)\}$

$\delta(q_0, a, a) = \{(q_0, \varepsilon)\}$

$\delta(q_0, b, b) = \{(q_0, \varepsilon)\}$

To simulate the above DPDA consider a string aab.

$\delta(q_0, aab, S) \vdash (q_0, aab, aS)$

$\vdash (q_0, ab, S)$

$\vdash (q_0, ab, aA)$

$\vdash (q_0, b, A)$.

$\vdash (q_0, b, b)$.

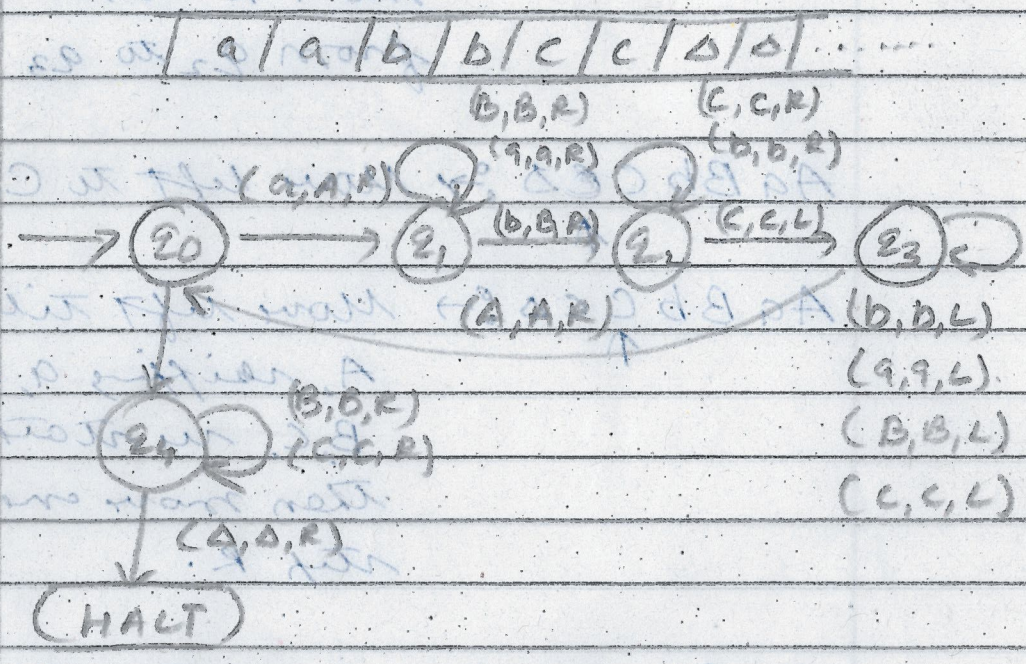
$\vdash (q_0, \epsilon)$.

Accept.

Qu-5 i)

ii) Construct a TM for $L = \{a^n b^n c^n / n \geq 1\}$

This problem cannot be solved even by PDA. We have solved this type of problem with turing machine as:



The simulation for aabbcc can be shown as below.

aabbccΔ :→ Convert a to A, move R transition from q_0 to q_1

AabbccΔ :→ Move R upto C.

AabbccΔ :→ convert b to B
↑ move right T.
q1 from q_1 to q_2

AaBbccΔ :→ Move R upto c.
↑

AaBbCcΔ :→ convert c to C,
↑ move R transition
from q_2 to q_3

AaBbCcΔ :→ Move left to C.
↑

AaBbCcΔ :→ Move left till
↑ A, skipping a, b,
B, C, in state q_3
then move one
step R.

AaBbCcΔ :→ convert a to A
↑ move right.

AaBbCcΔ :→ Move right
↑

AABbCcΔ :→ convert b to B
↑ and move R,
transition from
q_1 to q_2

AABBCC Δ $\overset{B}{\rightarrow}$ Move R.

AABBCC Δ $\overset{C}{\rightarrow}$ Convert c to C
move left till A,
refer q_3 state

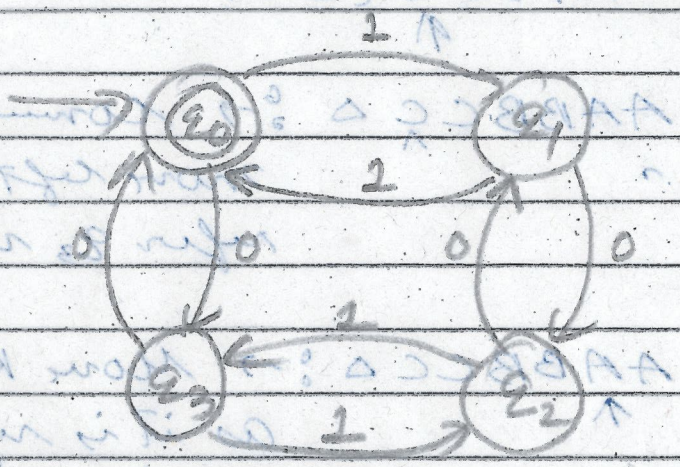
AABBCC Δ $\overset{A}{\rightarrow}$ Move R, keep B
as it is refer
transition from
q_0 to q_4.

AABBCC Δ $\overset{A}{\rightarrow}$ q_0 on moving
R by ignoring
B, C refer q_4 state

AABBCC Δ $\overset{\Delta}{\rightarrow}$ since Δ is read
TM will reach
to Halt state.

AABBCC Δ $\overset{\Delta}{\rightarrow}$

Qu-5 iii) For this type of problem
even we have drawn FSM.



And now it will be very much easy for us to convert the FSM to TM.