Even Semester Mid Term Examination, March 2024
Faculty of Engineering, School of Information, Security and Data Science
Department of Information Technology
B.Tech. – Information Technology

Course Code: IT3202	**Course: Automata Theory & Compiler Design**	**Semester: VI**
Time: 1.5 hrs.		**Max. Marks: 30**

Solution & Marking Scheme

SECTION A

S.No.		Marks
Q 1	Remember the concept of Arden's Theorem and construct a Regular Expression for the following FA which are shown in transition table. Here q1 is initial and final state.	2

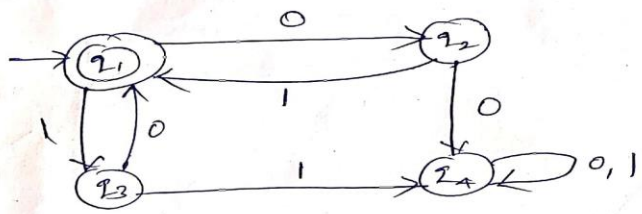
	0	1
→*q1	q2	q3
q2	q4	q1
q3	q1	q4
q4	q4	q4

Sol^n ① Arden's Theorem

Sol^n scheme

Complete Solution = 2 marks

Direct solution = 1 marks



$q_1 = q_2 \cdot 1 + q_3 \cdot 0 + \lambda$ ——— ①

$q_2 = q_1 \cdot 0$ ——— ②

$q_3 = q_1 \cdot 1$ ——— ③

$q_4 = q_2 \cdot 0 + q_3 \cdot 1 + q_4(0+1)$ ——— ④

Since q_1 is only final state; so as per arden's theorem we will solve only q_1 using ② & ③

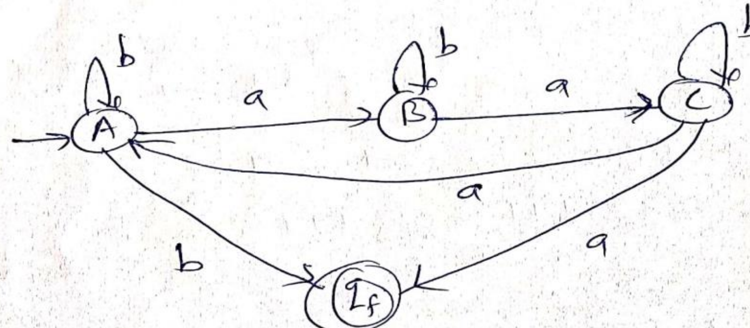
$q_1 = q_1 \cdot 0 \cdot 1 + q_1 \cdot 1 \cdot 0 + \lambda$

$\Rightarrow \underbrace{q_1}_{R} = \underbrace{\lambda}_{Q} + \underbrace{q_1}_{R}\underbrace{(01+10)}_{P}$

$\hookrightarrow R = QP^*$

$\therefore \boxed{q_1 = \lambda(01+10)^*}$

Ans

Q 2	Memorize the concept of regular grammar and convert following mentioned grammar into its equivalent Finite Automata. Sol^n ② ~~(b)~~) RG to FA $A \to aB \mid bA \mid b$ $B \to aC \mid bB$ $C \to aA \mid bC \mid a$ Let A, B, C are 3 states of desired FA  Sol^n Scheme Complete Sol^n ÷ 2 marks Sol^n without initial & final state ÷ 1.5 marks	**2**
Q 3	Recall the concept of left recursion, identify the left recursion in the following grammar and if so, remove it and rewrite the grammar. S→(T) | a , T→T;S | S Sol^n ③ <u>Left Recursion</u> ÷ Yes, ~~given~~ In given grammar, one of the production has left recursion $S \to (T) / a$ ← No left recursion $T \to T ; S / S$ ← left recursion ↓ solution We know that if $A \to A\alpha / \beta$ then $A \to \beta A'$ $A' \to \alpha A' / \epsilon$ Similarly $T \to ST'$ $T' \to ;ST' / \epsilon$ Final productions of grammar are :- $\boxed{S \to (T) / a \quad T \to ST' \quad T' \to ;ST' / \epsilon}$ Sol^n Scheme Identification : 0.5 marks Removal ÷ 1 marks Rewrite the grammar : 0.5 marks	**2**

SECTION B		
Q 4	Design a Mealy machine that gives 2's complement of any binary input. Assume that the last carry bit is neglected. Then, convert obtained Mealy machine into its equivalent Moore machine. Represent Mealy & Moore machine in both the form i.e. transition diagram and transition table as well.	**4**

Solution (4) Mealy & Moore Machine

2's Complement = 1's Complement + 1

e.g. MSB → 10100 ← LSB

01011 ← 1's complement

+1

01100 ← 2's complement

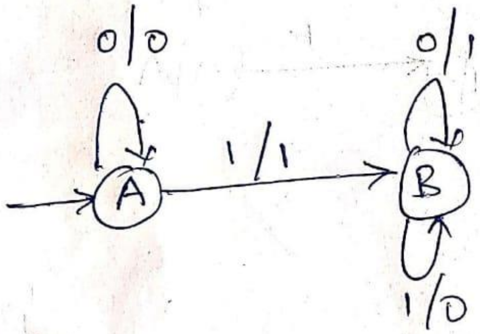
e.g. 11100 ← LSB

00011 ← 1's complement

+1

00100 ← 2's Complement

Logic: Occuring first 1 from LSB side, then its right side digits will be complemented



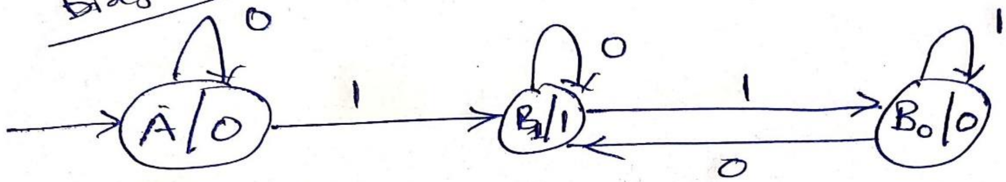
Mealy machine : Transition Diagram

States	0	1
→ A	A, 0	B, 1
B	B, 1	B, 0

Mealy Machine : Transition Table

Moore Machine

Moore Transition Diagram.



Transition Table

States	0	1	o/p
→ A	A	B_1	0
B_1	B_1	B_0	1
B_0	B_1	B_0	0

Solution Scheme: (solution 4)

For designing of Mealy M/C : 1.5 Marks
(Transition Diagram & Transition Table both)

For designing of Moore M/c : 2.5 Marks
(Conversion from Mealy to Moore)

Q5	Show that L = {**a**^{m} **b**^{m} **c**^{m} **| m > 0**} is not a Context Free Language using Pumping Lemma. Clearly mention all the desirable conditions for applying Pumping Lemma. Assume pumping length is 4.	**4**

Solution (5)

Pumping Lemma

⇒ Assume that language $L = \{a^n b^n c^n \mid n > 0\}$ is a context free language. Here, L have a pumping length $p = 4$.

⇒ Now, we take a string S such that $S = a^p b^p c^p$

⇒ As per pumping lemma, we divide S into 5 parts say u, v, x, y, z such that the following conditions must be true:

(i) $uv^i x y^i z$ is in L for every $i \geq 0$

(ii) $|vy| > 0$

(iii) $|vxy| \leq n$ where n is length of string $S = a^p b^p c^p$

Solution Scheme
Conditions : 1.5 marks
Apply Pumping Lemma : 2.5 marks

⇒ Now, we apply the conditions

$S = a^p b^p c^p = a^4 b^4 c^4$

⇒ $\underbrace{a}_{u}\underbrace{aa}_{v}\underbrace{abbbbc}_{x}\underbrace{c}_{y}\underbrace{cc}_{z}$

Conditions: $|vy| = |aac| = 3 > 0$

$|vxy| = |aaabbbbcy| = 9 \leq 12$ ← n

for $i = 2$ ⇒ $uv^2 x y^2 z$

⇒ $a(aa)^2 abbbbc(c)^2 cc$

⇒ $aaaaaabbbbccccc$

⇒ $a^6 b^4 c^5 \notin L$

Similarly, we can check for $i = 3$ ⇒ $a^8 b^4 c^6 \notin L$

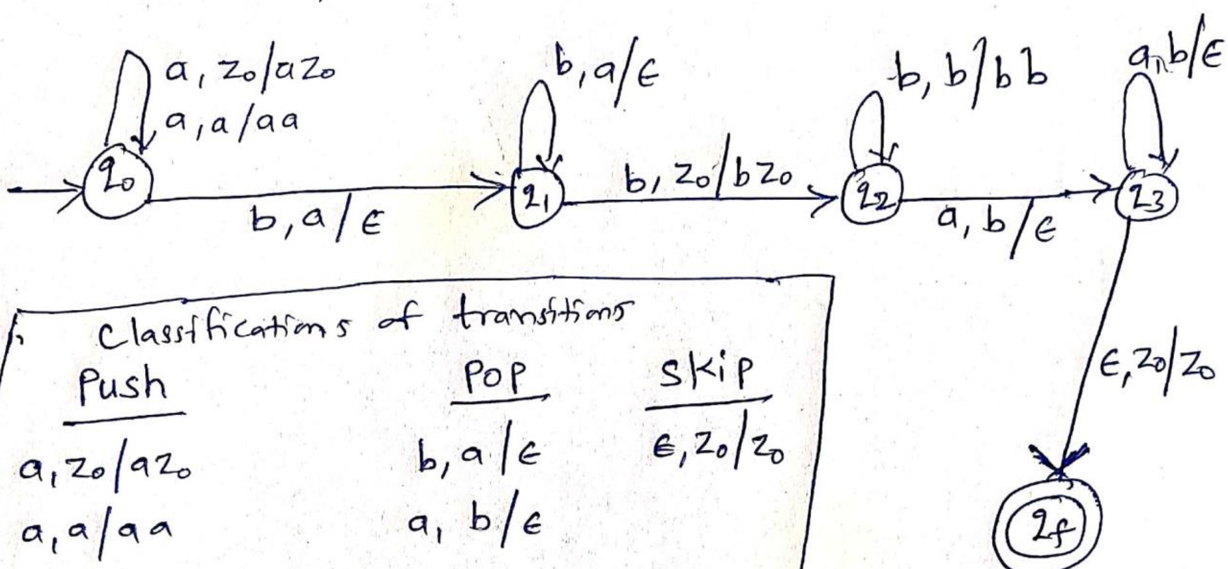
If rule violets, for any values of i, then we can say that given language is not context free language.

Q 6	Design a PDA for the language L = {$a^n\, b^{n+m}\, a^m$ | $n, m >= 1$}. Classify each transition of PDA with respect to different stack operations. Test your designed PDA by one of the valid strings and show PDA working step by step along with input tape and stack representation.	**4**

Sol^n (6) $L = \{ a^n b^{n+m} a^m \mid n, m \geq 1 \}$

Above language can be simplified as below ;

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



Classifications of transitions

Push	Pop	Skip
$a, z_0/az_0$	$b, a/\epsilon$	$\epsilon, z_0/z_0$
$a, a/aa$	$a, b/\epsilon$	
$b, z_0/bz_0$		
$b, b/bb$		

Solution Scheme : Designing of PDA : 1.5 marks

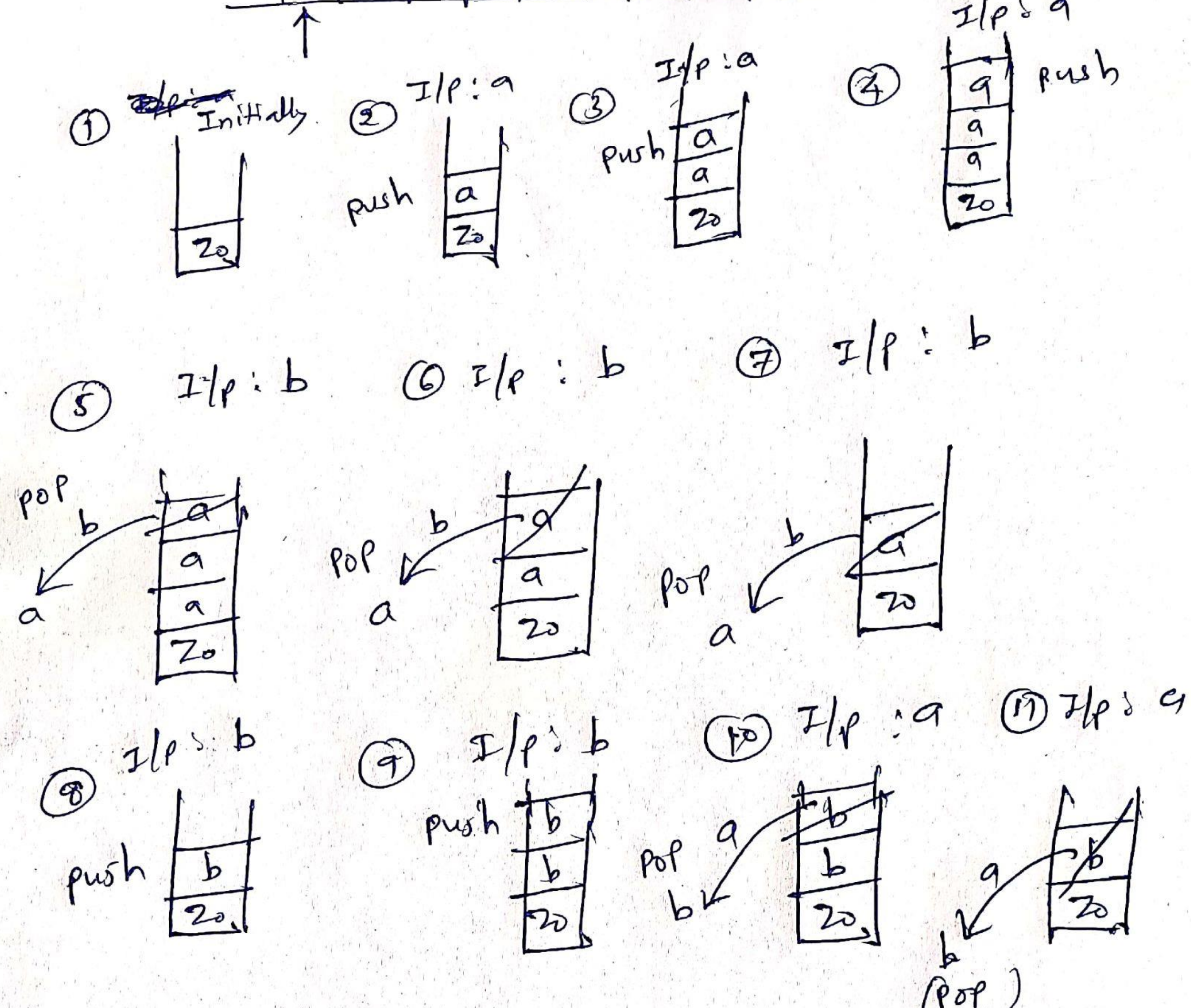
Classification of transitions : 1 marks

Testing of PDA through an valid string & stack : 1.5 marks

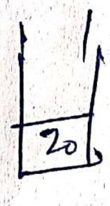
Sol^n (6) Testing of PDA by an valid string

Consider an ~~str~~ string $a^3 b^3 b^2 a^2$

a	a	a	b	b	b	b	b	a	a	ε



(12) Finally



Thus, testing is successful,

~~So, we have design a~~ valid PDA.

Q 7	Convert the following Context Free Grammar into Chomsky Normal Form S→ AACD A→ aAb / λ C→ aC / a D→ aDa / bDb / λ where λ is null string. Show each process of conversion step by step. Solution (7) ~~4(a)~~ CFG to CNF ① Elimination of Null productions S → AACD A → aAb / λ C → aC / a D → aDa / bDb / λ $\xRightarrow{\lambda\text{-removal}}$ S → AACD / ACD / CD / AAC / C / AC A → aAb / ab C → aC / a D → aDa / aa / bDb / bb ② Elimination of Unit Production S → AACD / ACD / CD / AAC / aC / a / AC A → aAb / ab C → aC / a D → aDa / bDb / aa / bb ③ Elimination of Useless Production :→ In above grammar, all the productions are useful; so no need to remove anyone. Solution scheme Simplification of CFG = 2 marks Conversion of CFG to CNF = 2 marks	**4**

Sol^n 7

CFG to CNF — P. 2

Now, grammar is simplified, We can start conversion from CFG to CNF as it requires two types of productions $\boxed{NT \rightarrow NT\ NT \text{ or } NT \rightarrow T}$ where NT = Non terminal & T = Terminal

~~Therefore~~, Let $J \rightarrow a$ & $K \rightarrow b$

replace J & K in appropriate position of grammar

$S \rightarrow AACD \mid ACD \mid CD \mid AAC \mid JC \mid a \mid AC$

$A \rightarrow JAK \mid JK$

$C \rightarrow JC \mid a$

$D \rightarrow JDJ \mid KDK \mid JJ \mid KK$

$J \rightarrow a$

$K \rightarrow b$

Above grammar is still not in CNF; so need some more replacement. Let $L \rightarrow AAC$; $M \rightarrow AC$; $N \rightarrow AA$; $O \rightarrow JA$; $P \rightarrow JD$ & $Q \rightarrow KD$

$S \rightarrow LD \mid MD \mid CD \mid NC \mid JC \mid a \mid AC$

$A \rightarrow OK \mid JK$

$C \rightarrow JC \mid a$

$D \rightarrow PJ \mid QK \mid JJ \mid KK$

$J \rightarrow a$

$K \rightarrow b$

$L \rightarrow AAC$

$M \rightarrow AC$

$N \rightarrow AA$

$O \rightarrow JA$

$P \rightarrow JD$

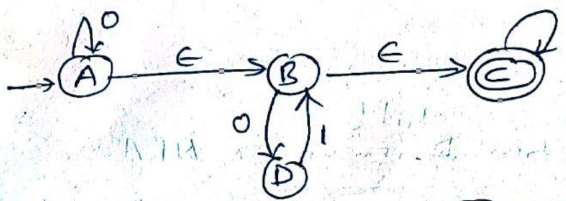
$Q \rightarrow KD$

Now this grammar is in CNF

	SECTION-C	
Q 8	Evaluate the following assignment by applying concept of finite automata:	**[5+3]**

(a) There are 4 bus stops in a highway namely A, B, C and D. According to a transport authority's agreement/policy. A Bus can move from 'A' to 'B' without paying any toll on the highway; similarly, bus can reach from 'A' to 'C' via 'B' without paying anything. But, from 'B' to 'D' and 'D' to 'B', bus has to pay in terms of '0' and '1' tokens respectively (decided in the policy by the authorities). In the same way, if bus moves from its own stop and still want to return to self then bus has to pay '0' and '1' token to reach stop from 'A' to 'A' and from 'C' to 'C' respectively. Simulate this situation with one of the finite automata models. Draw this scenario as transition diagram and transition table first. Then convert it into its equivalent NFA using Epsilon Closure method. Show all the computing steps in a tabular format for each entry of transition table corresponding to stops (assume as states) A, B, C and D against input {0, 1} of resultant NFA. Direct answer will reduce your marks accordingly.

Sol^n 8(a)



← Mapping into ε-NFA

state	I ε-closure(state)	II δ(state from I, Input)	III ε-closure (state of II)	δ(state, input)
A	{A, B, C}	δ(A,0) = A δ(B,0) = D δ(C,0) = φ	→ {A, B, C} → {D} → {φ}	Union → δ(A,0) = {A, B, C, D}
		δ(A,1) = φ δ(B,1) = φ δ(C,1) = C	→ {φ} → {φ} → {C}	union → δ(A,1) = {C}
B	{B, C}	δ(B,0) = D δ(C,0) = φ	→ {D} → {φ}	δ(B,0) = {D}
		δ(B,1) = φ δ(C,1) = C	→ {φ} → {C}	δ(B,1) = {C}
C	{C}	δ(C,0) = φ	{φ}	δ(C,0) = {φ}
		δ(C,1) = C	→ {C}	δ(C,1) = {C}
D	{D}	δ(D,0) = φ	→ {φ}	δ(D,0) = {φ}
		δ(D,1) = B	→ {B, C}	δ(D,1) = {B, C}

Transition ~~Diagram~~ Table for NFA

States	0	1
→ A	{A, B, C, D}	C
B	D	C
C	ϕ	C
D	ϕ	B, C

Solution Scheme :→

Mapping / simulation into ϵ-NFA = 1 marks

Calculation of ~~each~~ entry for A = 1.5 marks

entry for B = 1 marks

entry for C = 0.5 marks

entry for D = 0.5 marks

} 3.5

3.5 marks

ϵ-closure table

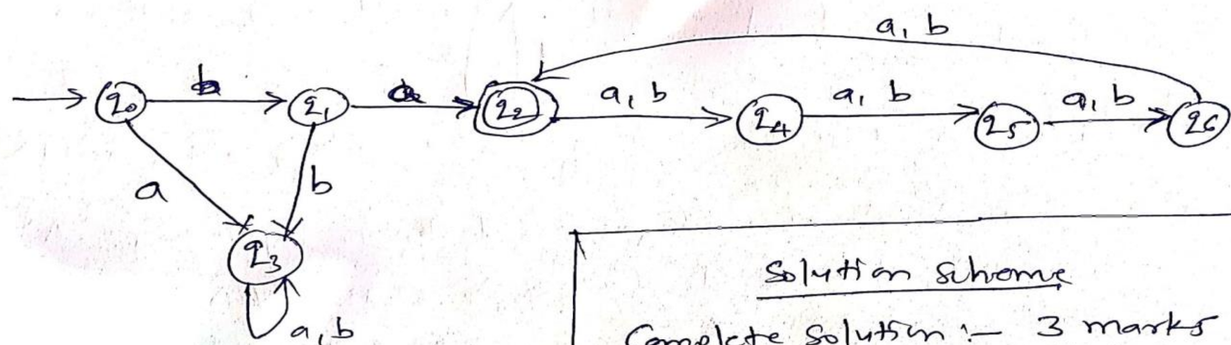
Final Transition Table for NFA = 0.5 marks

Direct Solution with ϵ-closure = 1.5 Marks

(b) Design a minimal DFA that accepts all strings over the alphabet {a, b} such that every accepted string ‘w’ should start with ‘ba’ and length is divisible by ‘2 (mod 4)’.

Solution Q (b)   ba & 2 (mod 4)

$|w| = 2, 6, 10, 14, 18$ ------ $\Sigma = \{a, b\}$



Solution Scheme

Complete Solution :— 3 marks

About to complete :— 2 marks

For logical & right efforts but partial complete — 1 marks