USN | | | | | | | | | | |

CMR INSTITUTE OF TECHNOLOGY, BENGALURU.

Internal Assessment Test 2 – Nov. 2017

Sub:	Automata Theory and Computability				Sub Code:	15CS54	Branch:	CSE,ISE
Date:	8/11/17	Duration:	90 min's	Max Marks:	50	Sem/Sec:	5/CSE(A,B,C)ISE(A)	OBE

	Answer any FIVE FULL Questions	MARKS	CO	RBT
1	Define a Context-free grammar (CFG). Write the CFG for the following languages. (a) L={$a^{2n}b^n$|n≥1} (b) L={$a^ib^jc^k$| j=i+k and i,k≥1} (c) L={w c w^R|w€ {a,b}*}	[10]	CO1 & CO3	L1 & L3
2	What is an ambiguous grammar? Show that the following grammar is ambiguous for the string **w=a+a^a**. Also write the equivalent unambiguous grammar for it. **E→E+E|E-E|E*E|E/E|(E)|E^E|a**	[10]	CO3	L1 & L3
3	Consider the following grammar. Generate the Leftmost derivation (LMD), Rightmost derivation (RMD), draw the LMD parse tree and RMD parse tree for the string **w=badbabaadb** . S →AaAb | BbBa A→aAb | bAB |d B→aB|bBa|ε	[10]	CO3	L3
4	Define Chomsky Normal Form (CNF). Convert the following CFG to CNF. S → ASA | aB A→B | S B→b | ε	[10]	CO4	L3
5 (a)	Define a PDA. Explain the working principle of PDA with a neat diagram.	[04]	CO4	L1
(b)	Design a PDA for the language L={a^nb^{2n}|n≥1}. Show the ID for the input string w=aabbbb	[06]	CO4	L3

USN | | | | | | | | | | |

CMR INSTITUTE OF TECHNOLOGY, BENGALURU.

Internal Assessment Test 2 – Nov. 2017

Sub:	Automata Theory and Computability				Sub Code:	15CS54	Branch:	CSE,ISE
Date:	8/11/17	Duration:	90 min's	Max Marks:	50	Sem/Sec:	5/CSE(A,B,C)ISE(A)	OBE

	Answer any FIVE FULL Questions	MARKS	CO	RBT
1	Define a Context-free grammar (CFG). Write the CFG for the following languages. (a) L={$a^{2n}b^n$|n≥1} (b) L={$a^ib^jc^k$| j=i+k and i,k≥1} (c) L={w c w^R|w€ {a,b}*}	[10]	CO1 & CO3	L1 & L3
2	What is an ambiguous grammar? Show that the following grammar is ambiguous for the string **w=a+a^a**. Also write the equivalent unambiguous grammar for it. **E→ E+E |E-E |E*E |E/E |(E) |E^E |a**	[10]	CO3	L1 & L3
3	Consider the following grammar. Generate the Leftmost derivation (LMD), Rightmost derivation (RMD), draw the LMD parse tree and RMD parse tree for the string **w=badbabaadb** . S →AaAb | BbBa A→aAb | bAB |d B→aB|bBa|ε	[10]	CO3	L3
4	Define Chomsky Normal Form (CNF). Convert the following CFG to CNF. S → ASA | aB A→B | S B→b | ε	[10]	CO4	L3
5 (a)	Define a PDA. Explain the working principle of PDA with a neat diagram.	[04]	CO4	L1
(b)	Design a PDA for the language L={a^nb^{2n}|n≥1}. Show the ID for the input string w=aabbbb	[06]	CO4	L3

6 (a)	Convert the following CFG to an equivalent empty stack PDA. S→aA A→aABC |bB | a B→b C→c	[05]	CO3 & CO4	L3
(b)	Define the following terms. (i) Language accepted by PDA (ii) Deterministic PDA	[05]	CO4	L1
7 (a)	Define NULL production, UNIT production and Useless symbol with example.	[03]	CO4	L1
(b)	Eliminate NULL production, Unit production and Useless symbols from the following grammar. S→ABCa | bD A→BC|b B→b| ε C→c|ε D→d	[07]	CO4	L3

6 (a)	Convert the following CFG to an equivalent empty stack PDA. S→aA A→aABC |bB | a B→b C→c	[05]	CO3 & CO4	L3
(b)	Define the following terms. (iii) Language accepted by PDA (iv) Deterministic PDA	[05]	CO4	L1
7 (a)	Define NULL production, UNIT production and Useless symbol with example.	[03]	CO4	L1
(b)	Eliminate NULL production, Unit production and Useless symbols from the following grammar. S→ABCa | bD A→BC|b B→b| ε C→c|ε D→d	[07]	CO4	L3

IAT - 2 , Nov. 2017

Sub: Automata Theory & Computability

Sub code: 15CS54

Scheme & Solution

(2M)

1. A context-free grammar (CFG) is a set of recursive rewriting rules (or productions) used to generate patterns of strings.

A CFG is defined by 4-tuples

$G = (V, T, P, S)$ or $G = (V, \Sigma, P, S)$

where V is a set of variables or nonterminals

T is a set of terminals

P is a set of productions

S is the start symbol.

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

$S \rightarrow aaSb \mid aab$

(b) $L = \{a^i b^j c^k \mid j = i+k \text{ and } i,k \geq 1\}$ (3M)

$a^i \; b^j \; c^k$

$a^i \; b^{i+k} \; c^k$

$\underbrace{a^i \; b^i}\; \underbrace{b^k \; c^k}$

$S \rightarrow AB$

$A \rightarrow aAb \mid ab$

$B \rightarrow bBc \mid bc$

(c) $L = \{wcw^R \mid w \in \{a,b\}^*\}$ (3M)

$S \rightarrow aSa \mid bSb \mid c$

2. <u>Ambiguous grammar</u> (1M)

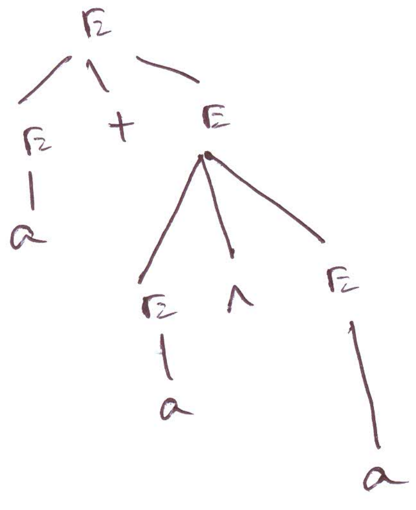
A CFG is said to be ambiguous if there exist a string which has more than one LMD or more than one RMD.

Given grammar

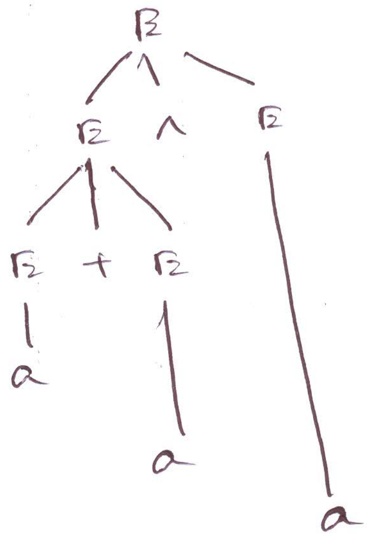
$E \rightarrow E+E \mid E-E \mid E*E \mid E/E \mid (E) \mid E \wedge E \mid a$

$w = a+a \wedge a$ (5M)

<u>LMD1</u>



<u>LMD2</u>



Yes. It is ambiguous.

<u>Unambiguous grammar</u> (4M)

$E \rightarrow E+T \mid E-T \mid T$

$T \rightarrow T*F \mid T/F \mid F$

$F \rightarrow G \wedge F \mid G$

$G \rightarrow (E) \mid a$

The given grammar is

$S \to AaAb \mid BbBa$

$A \to aAb \mid bAB \mid d$

$B \to aB \mid bBa \mid \epsilon$

1

$w = badbabaadb$

(3+2) M

LMD

$S \Rightarrow AaAb$

$\Rightarrow bABaAb$ $(A \to bAB)$

$\Rightarrow baAbBaAb$ $(A \to aAb)$

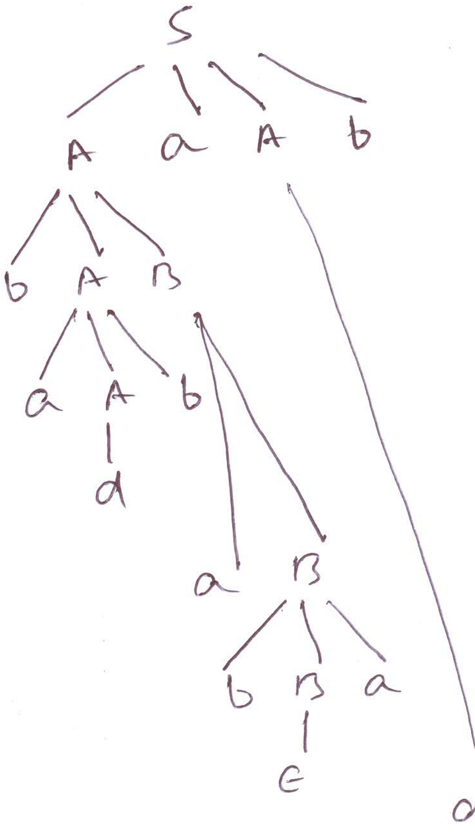
$\Rightarrow badbBaAb$ $(A \to d)$

$\Rightarrow badbaBaAb$ $(B \to aB)$

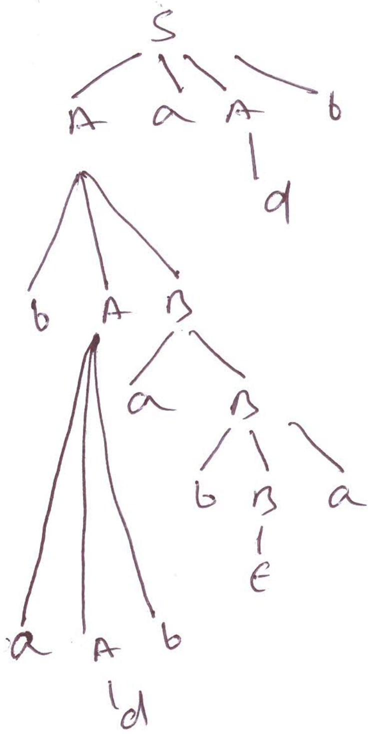
$\Rightarrow badbabBaaAb$ $(B \to bBa)$

$\Rightarrow badbab\epsilon aaAb$ $(B \to \epsilon)$

$\Rightarrow badbabaadb$ $(A \to d)$



RMD



(3+2) M

$S \Rightarrow AaAb$

$\Rightarrow Aadb$ $(A \to d)$

$\Rightarrow bABadb$ $(A \to bAB)$

$\Rightarrow bAaBadb$ $(B \to aB)$

$\Rightarrow bAabBaadb$ $(B \to bBa)$

$\Rightarrow bAab\epsilon aadb$ $(B \to \epsilon)$

$\Rightarrow baAbabaadb$ $(A \to aAb)$

$\Rightarrow badbabaadb$ $(A \to d)$

4. Chomsky Normal Form (CNF)

A context-free grammar G is said to be in CNF if all of its productions rules are of the form (2M)

$A \to BC$ or

$A \to a$

where A, B, C are non terminal symbols, a is a terminal symbol.

Given CFG is

$S \to ASA \mid aB$

$A \to B \mid S$

$B \to b \mid \epsilon$

Step 1

Remove ϵ-prod^n

Nullable set = $\{A, B\}$ (2M)

$S \to ASA \mid aB \mid AS \mid SA \mid S \mid a$

$A \to B \mid S$

$B \to b$

Step 2

Remove Unit prod^n (2M)

$S \to ASA \mid aB \mid AS \mid SA \mid a$

$A \to b \mid ASA \mid aB \mid AS \mid SA \mid a$

$B \to b$

step 3

No useless symbols (2M)

step 4

Convert to CNF (2M)

$S \rightarrow XA \mid YB \mid AS \mid SA \mid a$

$X \rightarrow AS$

$Y \rightarrow a$

$A \rightarrow b \mid XA \mid YB \mid AS \mid SA \mid a$

$B \rightarrow b$

(4M)

5.(a) A pushdown automaton is a way to implement a context-free grammar in a similar way we design DFA for a regular grammar. A DFA can remember a finite amount of information, but a PDA can remember an infinite amount of information.

A PDA can be formally described as a 7-tuple $(Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$

where Q is finite set of states

Σ is I/P alphabet

Γ is stack symbols

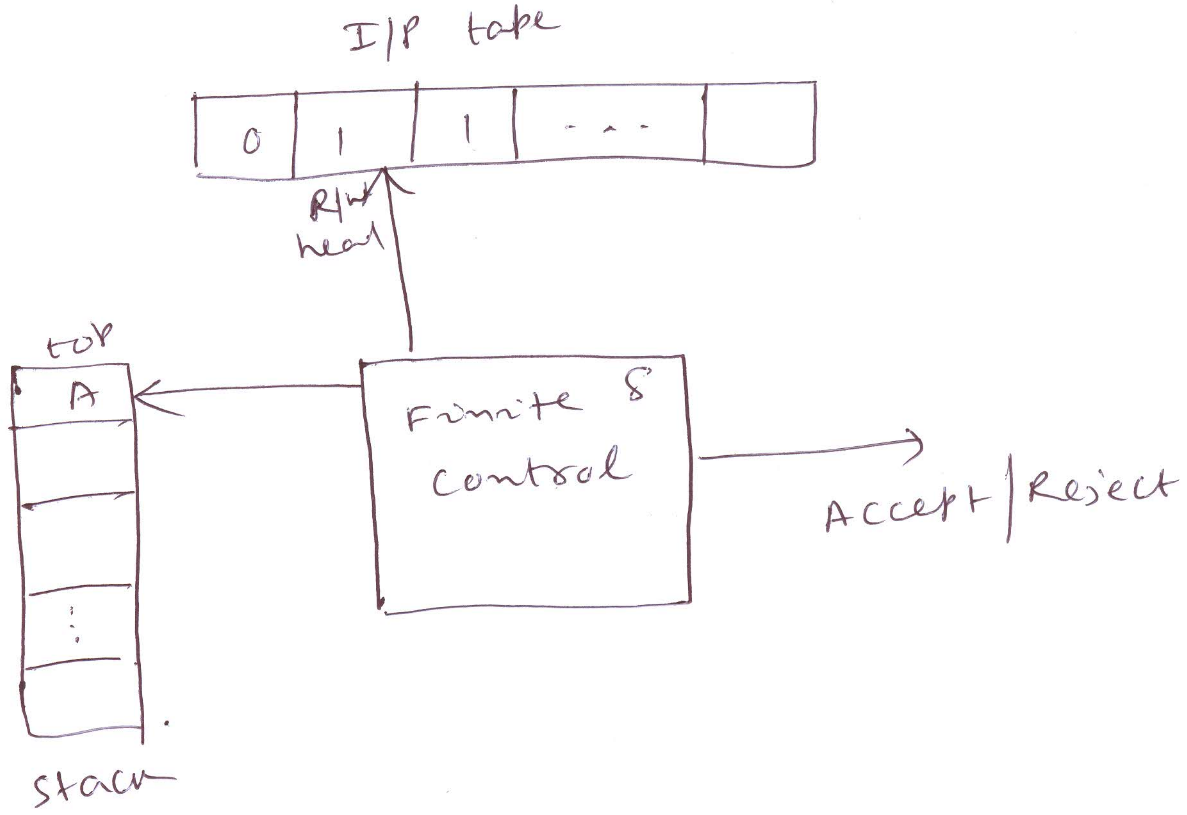
δ is transition function : $Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \rightarrow Q \times \Gamma^*$

q_0 initial state

z_0 stack top symbol

F " final state.

Basic Block diagram

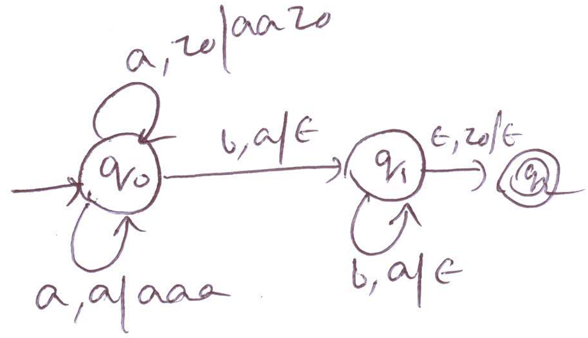


A PDA has 3 components -

1. An input tape
2. A control unit
3. A stack with infinite size

5. (b) $L = \{a^n b^{2n} \mid n \geq 1\}$ (3M + 3M)

Method 1



$\delta(q_0, a, z_0) = (q_0, aaz_0)$

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

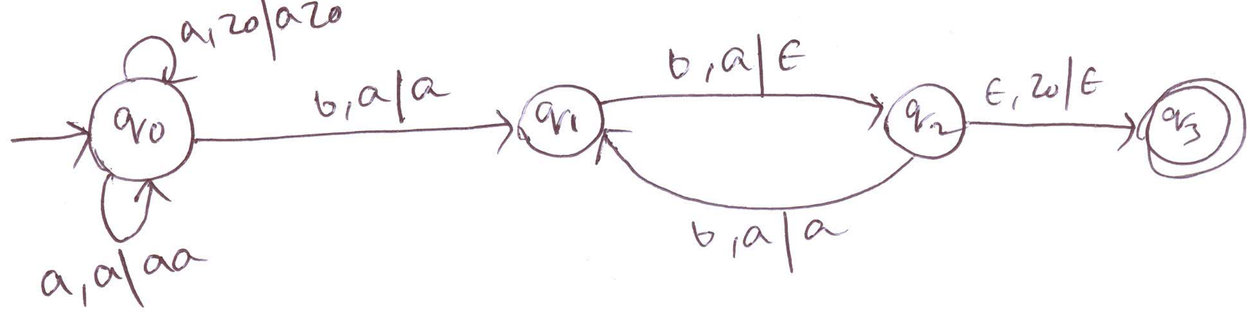
$\delta(q_0, b, a) = (q_1, \epsilon)$

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

$\delta(q_1, \epsilon, z_0) = (q_2, \epsilon)$

Method 2

Ex aabbbb



$$\delta(q_0, a, Z_0) = (q_0, aZ_0)$$

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

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

$$\delta(q_1, b, a) = (q_2, \epsilon)$$

$$\delta(q_2, b, a) = (q_1, a)$$

$$\delta(q_2, \epsilon, Z_0) = (q_3, \epsilon)$$

ID for $w = aabbbb$ using method 1

$(q_0, aabbbb, Z_0) \vdash (q_0, abbbb, aaZ_0)$

$\vdash (q_0, bbbb, aaaaZ_0)$

$\vdash (q_1, bbb, aaaZ_0) \vdash (q_1, bb, aaZ_0)$

$\vdash (q_1, b, aZ_0) \vdash (q_1, \epsilon, Z_0)$

$\vdash (q_2, \epsilon, \epsilon)$ <u>Accepted</u>

ID for $w = aabbbb$ using Method 2

$(q_0, aabbbb, Z_0) \vdash (q_0, abbbb, aZ_0) \vdash (q_0, bbbb, aaZ_0)$

$\vdash (q_1, bbb, aaZ_0) \vdash (q_2, bb, aZ_0)$

$\vdash (q_1, b, aZ_0) \vdash (q_2, \epsilon, Z_0)$

$\vdash (q_3, \epsilon, \epsilon)$

Accepted

(5M)

6. (a) $\delta(q, \epsilon, S) = (q, aA)$

$\delta(q, \epsilon, A) = \{(q, aABC), (q, bB), (q, a)\}$

$\delta(q, \epsilon, B) = (q, b)$

$\delta(q, \epsilon, C) = (q, c)$

$\delta(q, a, a) = (q, \epsilon)$

$\delta(q, b, b) = (q, \epsilon)$

$\delta(q, c, c) = (q, \epsilon)$

(b) Language accepted by PDA (2.5)

There are two different ways to define PDA acceptability

1. Final state PDA

$L(PDA) = \{ w \mid (q_0, w, Z_0) \vdash^* (q, \epsilon, x), q \in F \}$

② Empty stack PDA

$L(PDA) = \{ w \mid (q_0, w, z_0) \vdash^* (q, \epsilon, \epsilon), q \in Q \}$

Deterministic PDA (2.5M)

A PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$ to be deterministic iff the following conditions are met.

1. $\delta(q, a, X)$ has at most one member for any q in Q, a in Σ or $a = \epsilon$ and X in Γ.

2. If $\delta(q, a, X) \neq \phi$, then $\delta(q, \epsilon, X) = \phi$

(3M)

7. (a)

NULL Production

In a CFG, a non-terminal 'A' is a nullable variable if there is a production $A \to \epsilon$ or there is a derivation that starts at A and finally ends up with

$A \to \text{------} \to \epsilon$

Ex $S \to aA \mid b$
$A \to \epsilon$

Unit production

Any production rule in the form $A \to B$ is called unit prod^n. where A, B are non termin

Useless symbol

Those symbols that do not participate in derivation of any string is called as the useless symbols.

Ex $S \to Ab$

$A \to a$

$B \to b$

Here B is useless.

(b) Given grammar is (3+3+1) Mark

$S \to ABCa \mid bD$

$A \to BC \mid b$

$B \to b \mid \epsilon$

$C \to c \mid \epsilon$

$D \to d$

Step1 : Remove ϵ-prod^n

Nullable set = $\{A, B, C\}$

$S \to ABCa \mid bD \mid BCa \mid ACa \mid ABa \mid Ca \mid Aa \mid Ba \mid a$

$A \to BC \mid b \mid B \mid C$

$B \to b$

$C \to c$

$D \to d$

step2 : Remove Unit prodⁿ

$S \to ABCa \mid bD \mid BCa \mid ACa \mid ABa \mid Ca \mid Aa \mid Ba \mid a$

$A \to BC \mid b \mid c$

$B \to b$

$C \to c$

$D \to d$

step3 : Remove Useless symbol

No useless symbol.