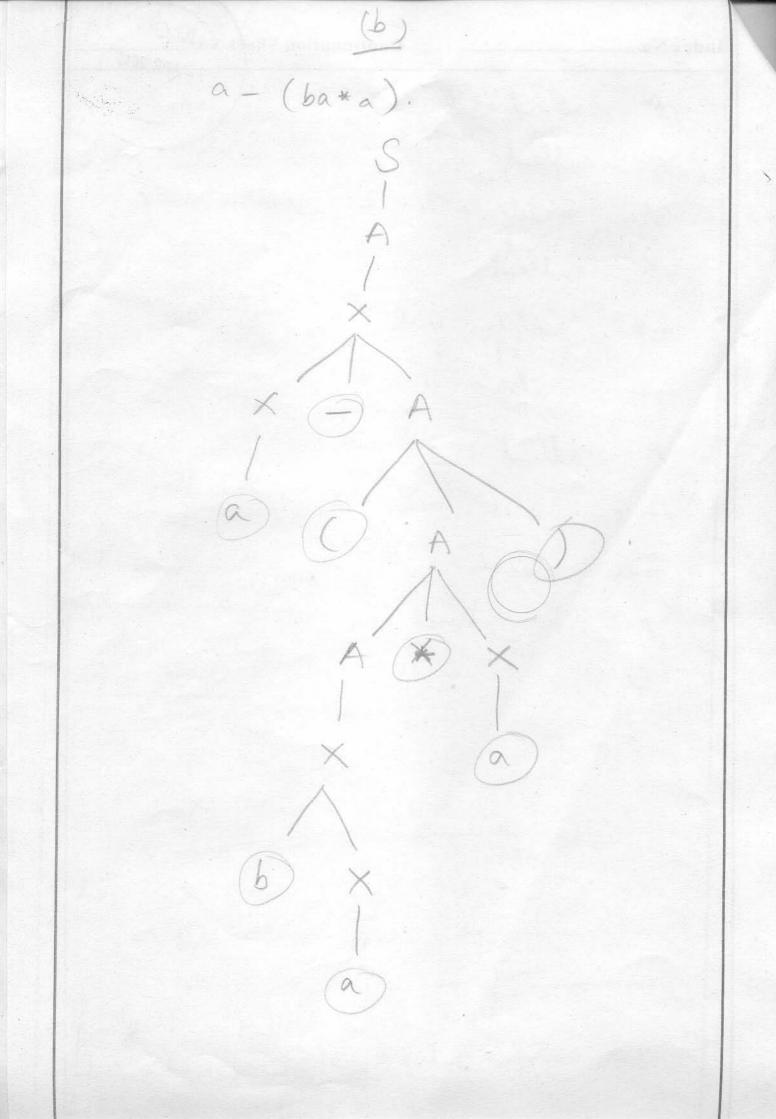
15 APR 2010 Continuation Sheet No. 3 Index No .-Q#2 bool S() if (A()) return true; else return Jake; bool A() if ( token = = '(') && A() && son token == ')') return true else if (AL) and && token == \* (& XU) return true; else if (XU) Setuen True, else retuen . Jalse;

bool x() 3 if (x1) &8 +8ken == '- ' &8 A()) Leturn true; else if (tolen = = 'a') Letren true; else it (token == 16' 12 ×()) return thre; else setuin Jake; imput: a - (ba+ a) parsing order S called. tota worth with a return true. >X // (X-A) tolen match with a, return true toku wolds with (, solur this

Continuation Sheet No. N. CS 15 APR 2010 Index No ... A called ( A \* X token mælch with b, Return there called return true token watch with a token match " \*, X called - tolen welch with a return the tolen match with ) A vetuen true



	/ MC
Index No.	Continuation Sheet No. 15 APR 2010
C	Pakistan
	> < EXPR> < ARITH_OPR> < EXP
	CUNARY_OPR>CIDENT>
	IDENT> (RIGHT_UNARY_OPR)
	CTERM>
(ARITH_OPR) ->	+   -   *   /   %
< LEFT UNARY_ OPR > _	-> + ++ ~
< RIGHT_UNARY_OPR>	
< EXPR>>	CARITH_EXAR> CLOGICAL - EXA
1	CRELAT_EXPR> (CEXPR>)
< TERM >	<literal> &lt; FUN_CALL&gt;</literal>
CLITERALY > C	INT CONSTO   CHARCONSTO
	FLOAT_CONSTD (STRING_CONST)
(FUN_CALL) ->	CIDENT> (CPARAM>)

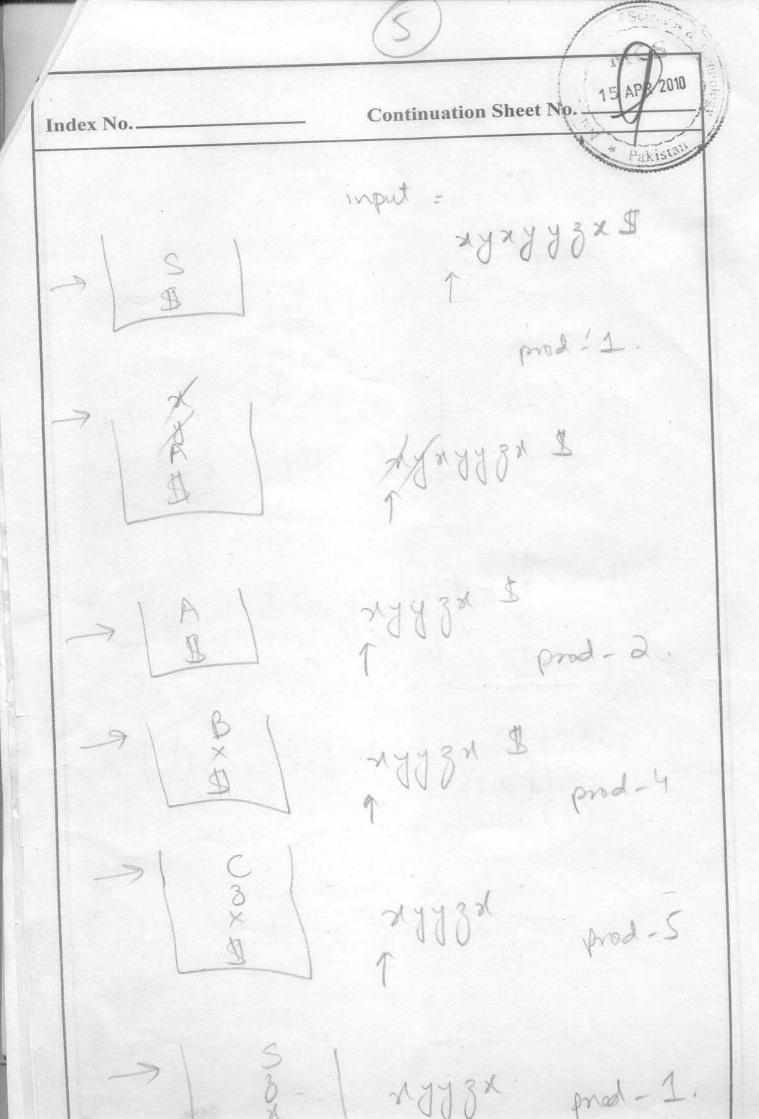
PARAMY

Start end EXPR ARITH--> EXPR Unary ->[IDENT] RDENT > RIGHT A TERM ARITH-DPR Start

Continuation Sheet No. 15 Index No. \_\_\_ Ceft\_ Unaly - Opr Right - Unaly - Opr start

(G): 3 (9) 1. S -> xy A 3 x 3 2. A -> Bx {x,3} y ← A. E 到了. 4. B -> Cg 3x,33 5. C -> S 3 x g 6. C -> 3 183

Yes it is LL(1)



Lyzx & y 3x & prod-3 Successful? Parsing.

Index No.

Continuation Sheet No.

15 APR 2010

Pakistan

b

S -> a = AB; LL(0)

A -> B C

B -> aC | bA - LL(1)

C -> E | BC - LL (1)

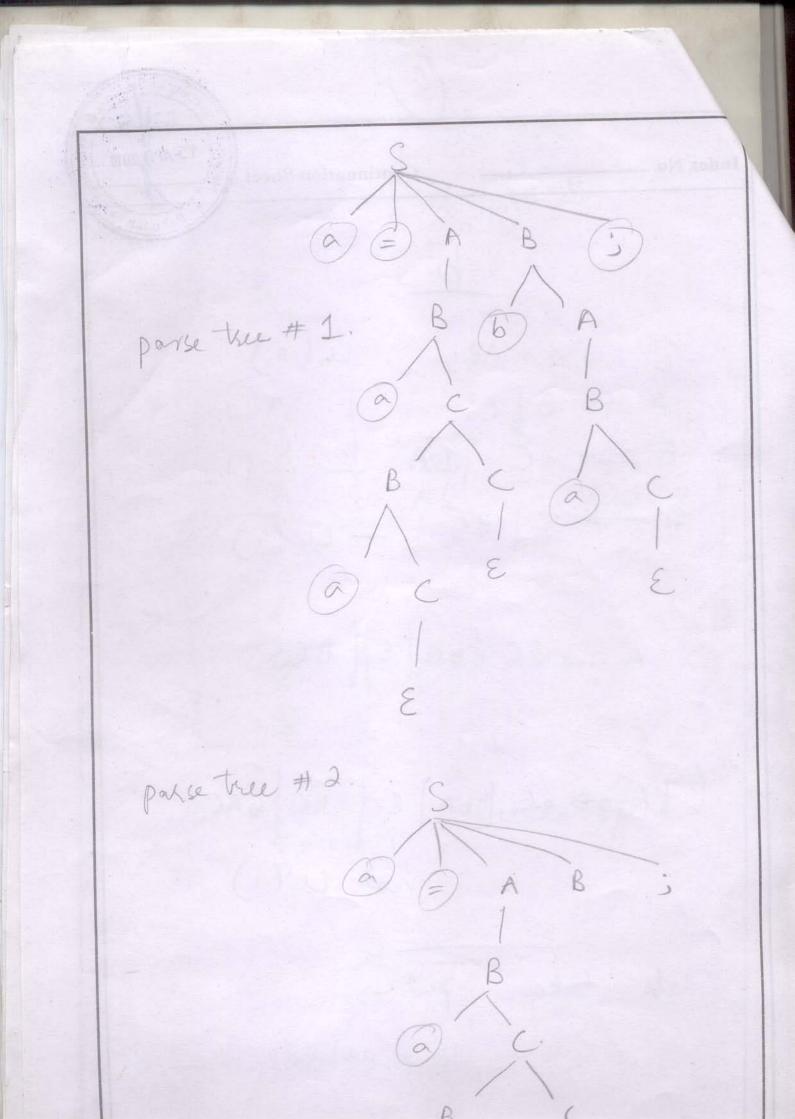
A -> aC | bA | E | BC

V

A -> aC | bA | E | acc | bAC.

lots taken input as

a = aaba;



MCS 15 APR 2010 Continuation Sheet No. Index No. \_ Yes et d'ambigous Not left-reculsive Lame symbol or LHS gince and R. H.S Saga ABi X CLIAL E BC