Grammar Lassign>::=Lid7=Lexpr> 212) ::= A | B | C Lexpr):= Lid) + Lexpr) (Lid) \* Lexpr) (Lexpr)) 2:2) (assign) Lassign):= Rexpr := Lid> +lexpr> (C \* A)) == A \* Lexpr) := A \* (Lexpr) ::= A \*(Lid)+(expr)) :== A # (B+ Lexpr) :: = A \* (B + (Xexpr)) 220 1: = A \* (B+(C \* Lexpr)) ::= A \* (B+(C \* <id))) := A\*(B+(C+A))

b) B=C\*(A\*c+B) Lidi= Lexpr) Lidos= A/B/C (cxpr):= (id) (Lexpr)) / 2:87 + Lexpr) / 4:87 + Lexpr) Lassign) 2:15 \* Zid) Zexpr) (Assign) i= Lid = Lexpr) A \* C + Lexper B-> Zexpr > Sody (expr) > C \$ Lepr> > (\*(Lepr>) -> C \* (<id) \* Lexer) DC\* (A + Cid)+ Lexer) ->C \*(A\*C+ <id>) -> C \*(A \* C+ B)

w=aabab S-> aB/bA 30 as 6AA a B-> 65/a 88/6 right most SDas ->a a B -> a a b S DaabaB Da abab Jeft most S> aB -> aaBB - aabs B -> a abab -) aabab

Contains Strings with chara \$60 21 char each with char 'c' being the terminating char (S) > (A) > aSb ab A B A B ab b ab b ab b ab23->6 Leroy > bogin Lotust arr ) end (Stuf and ) Stuf E) Btuf NO3 (8+mb) -> (var) = (expr) (Var) -> ABC Lexpros War> (Lvar) Lvar) 6 5-) AB B -> 6AB/6A A -> af abA