program.trans = list, trans. program -> MAIN LRAREN RRAREN LCURLY list RCURLY list1. trans = list2, trans U one I tem. trans list - list one Item list . trans = {} list -> epsilon one I tem. trans = }} one I tem -> decl one Item . trans = stmt, trans one I tem -> stmt ded -> BOOL ID SEMICOLON ded - INT ID SEMICOLON {} stmt -> ID ASSIGN exp SEMICOLON stmt, trans = espitroins. Start -> IF LRAKEN EXP RPAREN SEARCE stml1. trans = exp. trans Ustmt2, trans Stat -> WHILE LRAREN exp RPAREN start start. troops = exp. troops U startz. troops. Stint -> LCURLY list RCURLY semt1. trans = list. trains. exp. trans = expz. trans U exp3 trans exp -> exp TIMES exp exp -> exp DIVIDE exp expitrans = expitrans U exps. trans expitraris = expitrans Vesps. trans. exp -> exp PLUS exp esp -> Kesp LESS esp expitrans=expitrans Vexpitrans. exp -> exp EQUAL exp expitrans = expitrons Vespitrans. EXP -> # LARTH EXP RPAREN expitrans = expitrans. exp -> ID exp. trans = {} ENP -> BOOLLITERAL esp, trans =  $\{\}$ exp -> INTLITERAL exp. froms = { INTLITERAL, value }