Nonterminal	VAR	::	;	Integer	Bool	(	)	"="	if	then	else	"=="	<=	"+"	-	NUM	BOOLEAN	(Exp)	->	$
Prog	Prog -> Decl Prog																			Prog -> "
Decl	Decl -> TypeDecl TermDecl																			
TypeDecl	TypeDecl -> VAR :: Type ;																			
Type				Type -> Type0 TypeRest	Type -> Type0 TypeRest	Type -> Type0 TypeRest														
TypeRest			TypeRest -> "				TypeRest -> "												TypeRest -> -> Type	
Type0				Type0 -> Integer	Type0 -> Bool	Type0 -> (Type)														
TermDecl	TermDecl -> VAR Args = Exp ;																			
Args	Args -> VAR Args							Args -> "												
Exp	Exp -> Exp0								Exp -> if Exp then Exp else Exp							Exp -> Exp0	Exp -> Exp0	Exp -> Exp0		
Exp0	Exp0 -> Exp1 Rest0															Exp0 -> Exp1 Rest0	Exp0 -> Exp1 Rest0	Exp0 -> Exp1 Rest0		
Rest0			Rest0 -> "				Rest0 -> "			Rest0 -> "	Rest0 -> "	Rest0 -> == Exp1	Rest0 -> <= Exp1							
Exp1	Exp1 -> Exp2 Rest1					Exp1 -> Exp2 Rest1										Exp1 -> Exp2 Rest1	Exp1 -> Exp2 Rest1	Exp1 -> Exp2 Rest1		
Rest1			Rest1 -> "				Rest1 -> "			Rest1 -> "	Rest1 -> "	Rest1 -> "	Rest1 -> "	Rest1 -> + Exp2 Rest1	Rest1 -> - Exp2 Rest1					
Exp2	Exp2 -> Exp3 Rest2					Exp2 -> Exp3 Rest2										Exp2 -> Exp3 Rest2	Exp2 -> Exp3 Rest2	Exp2 -> Exp3 Rest2		
Rest2	Rest2 -> Exp3 Rest2		Rest2 -> "				Rest2 -> "			Rest2 -> "	Rest2 -> "	Rest2 -> "	Rest2 -> "	Rest2 -> "	Rest2 -> "	Rest2 -> Exp3 Rest2	Rest2 -> Exp3 Rest2	Rest2 -> Exp3 Rest2		
Exp3	Exp3 -> VAR					(Type)										Exp3 -> NUM	Exp3 -> BOOLEAN	Exp3 -> (Exp)		