## C-- Grammar:

Program ------> VarDeclList FunDeclList

VarDeclList --> **epsilon**

VarDecl VarDeclList

VarDecl ------> Type **id** VarDecl`

VarDecl`------> **;**

**[ num] ;**

FunDeclList --> FunDecl FunDeclList`

FunDeclList` --> **epsilon**

FunDeclList

FunDecl ------> Type **id** **(** ParamDecList **)** Block

ParamDeclList --> **epsilon**

ParamDeclListTail

ParamDeclListTail --> ParamDecl ParamDeclListTail`

ParamDeclListTail` --> **epsilon**

**,** ParamDeclListTail

ParamDecl ----> Type **id** ParamDecl`

ParamDecl`----> **epsilon**

**[]**

Block --------> **{** VarDeclList StmtList **}**

Type ---------> **int**

**char**

StmtList -----> Stmt StmtList`

StmtList` ----> **epsilon**

StmtList

Stmt ---------> **;**

Astmt **;**

**return** Expr **;**

**read** **id** **;**

**write** Expr **;**

**writeln** **;**

**break** **;**

**if** **(** Expr **)** Stmt **else** Stmt

**while** **(** Expr **)** Stmt

Block

Astmt ---> **id** Expr!

Expr ---------> Primary Expr’

UnaryOp Expr Expr’

**id** Expr!

Expr! -----> **=** Expr Expr’

**[** Expr **]** **=** Expr Expr’

Expr’ -------> **epsilon**

BinOp Expr Expr’

Primary ------> **id** Primary`

**num**

**(** Expr **)**

Primary` -----> **epsilon**

**(** ExprList **)**

**[** Expr **]**

ExprList -----> **epsilon**

ExprListTail

ExprListTail --> Expr ExprListTail`

ExprListTail` --> **epsilon**

**,** ExprListTail

UnaryOp ------> **-** **|** **!**

BinOp --------> **+** **|** **-** **|** **\*** **|** **/ |** **==** **|** **!=** **|** **<** **|** **<=** **|** **>** **|** **>=** **|** **&&** **|** **||**