**Grammear G**

$

int float void

int float void

int float void Id { if return

int float void Id { if return

id

]

]

int float void

int float void

; {

$

$

int float void $

id

)

)

)

, )

, )

int float void ; $ }

Id { if return

Id { if return

}

}

; }

; }

; }

)

int float

int float

int float

int float

; [

int float

int\_num

,

int float void

int float void

int float void

int float void

int float void

int float void

int float void

int float

int float

,

int float

[

{

int float

int float

Id { if return

;

Id { if return

[ ( =

if

id

id int\_num float\_num (

\*PROG -> GLOBAL\_VARS FUNC\_PREDEFS FUNC\_FULL\_DEFS

\*GLOBAL\_VARS -> VAR\_DEC GLOBAL\_VARS’

\*GLOBAL\_VARS’ -> VAR\_DEC GLOBAL\_VARS’ | Ɛ

\*VAR\_DEC –> TYPE id VAR\_DEC’

\*VAR\_DEC’ -> ; | [ DIM\_SIZES ] ;

\*TYPE -> int | float

\*DIM\_SIZES -> int\_num DIM\_SIZES’

\*DIM\_SIZES’ -> Ɛ | , DIM\_SIZES

\*FUNC\_PREDEFS -> FUNC\_PROTOTYPE ; FUNC\_PREDEFS’

\*FUNC\_PREDEFS’ -> FUNC\_PROTOTYPE ; FUNC\_PREDEFS’ | Ɛ

\*FUNC\_PROTOTYPE -> RETURNED\_TYPE id (PARAMS)

\*FUNC\_FULL\_DEFS -> FUNC\_WITH\_BODY FUNC\_FULL\_DEFS’

\*FUNC\_FULL\_DEFS’ -> FUNC\_FULL\_DEFS | Ɛ

\*FUNC\_WITH\_BODY -> FUNC\_PROTOTYPE COMP\_STMT

\*RETURNED\_TYPE -> TYPE | void

\*PARAMS -> PARAM\_LIST | Ɛ

\*PARAM\_LIST -> PARAM PARAM\_LIST’

\*PARAM\_LIST’ -> , PARAM PARAM\_LIST’ | Ɛ

\*PARAM –> TYPE id PARAM’

\*PARAM’ -> Ɛ | [ DIM\_SIZES ]

\*COMP\_STMT -> { VAR\_DEC\_LIST STMT\_LIST }

\*VAR\_DEC\_LIST -> Ɛ VAR\_DEC\_LIST’

\*VAR\_DEC\_LIST’ -> VAR\_DEC VAR\_DEC\_LIST’ | Ɛ

\*STMT\_LIST -> STMT STMT\_LIST’

\*STMT\_LIST’ -> ; STMT STMT\_LIST’ | Ɛ

\*STMT -> id STMT’ | COMP\_STMT | IF\_STMT | RETURN\_STMT

\*STMT’ -> VAR’ = EXPR | (ARGS)

\*IF\_STMT -> if ( CONDITION ) STMT

\*CALL -> id ( ARGS )

\*ARGS -> ARG\_LIST | Ɛ

\*ARG\_LIST -> EXPR ARG\_LIST’

)

)

; }

; }

= \* + rel\_op ) , ; } ]

]

]

)

rel\_op ) , ; } ]

rel\_op ) , ; } ]

+ rel\_op ) , ; } ]

+ rel\_op ) , ; } ]

\* + rel\_op ) , ; } ]

\* + rel\_op ) , ; } ]

id int\_num float\_num (

,

return

id int\_num float\_num (

id

[

id int\_num float\_num (

,

id int\_num float\_num (

id int\_num float\_num (

+

id int\_num float\_num (

\*

id int\_num float\_num (

( [

\*ARG\_LIST’ -> , EXPR ARG\_LIST’ | Ɛ

\*RETURN\_STMT -> return RETURN\_STMT’

\*RETURN\_STMT’ -> Ɛ | EXPR

\*VAR -> id VAR’

\*VAR’ -> Ɛ | [ EXPR\_LIST ]

\*EXPR\_LIST -> EXPR EXPR\_LIST’

\*EXPR\_LIST’ -> , EXPR EXPR\_LIST’ | Ɛ

\*CONDITION -> EXPR rel\_op EXPR

\*EXPR -> TERM EXPR’

\*EXPR’ -> + TERM EXPR’ | Ɛ

\*TERM -> FACTOR TERM’

\*TERM’ -> \* FACTOR TERM’ | Ɛ

\*FACTOR -> id FACTOR’ | int\_num | float\_num | ( EXPR )

\*FACTOR’ -> VAR’ | (ARGS)

**Nullable Not Nullable**

**First Follow**