

## Gramática

### PRODUCCIÓN

programa → declaraciones funciones  
declaraciones → tipo lista\_var ; declaraciones | ε  
tipo → basico compuesto  
basico → **int** | **float** | **char** | **double** | **void**  
compuesto → ( **numero** ) compuesto | ε  
lista\_var → lista\_var , **id** | **id**  
funciones → **func** tipo **id** ( argumentos ) bloque funciones | ε  
argumentos → lista\_args | ε  
lista\_args → lista\_args , tipo **id** | tipo **id**  
bloque → { declaraciones instrucciones }  
instrucciones → instrucciones sentencia | sentencia  
sentencia → parte\_izquierda = bool ; **if**( bool ) sentencia  
| **if**( bool ) sentencia **else** sentencia | **while**( bool ) sentencia  
| **do** sentencia **while**( bool ) | **break** ; | bloque | **return** exp ; | **return** ;  
| **switch**( bool ) { casos } | **print** exp ; | **scan** parte\_izquierda  
casos → caso casos | ε | predeterminado  
caso → **case** **numero** : instrucciones  
predeterminado → **default** : instrucciones  
parte\_izquierda → **id** localizacion | **id**  
bool → bool || comb | comb  
comb → comb && igualdad | igualdad  
igualdad → igualdad == rel | igualdad != rel | rel  
rel → exp < exp | exp <= exp | exp >= exp |  
| exp > exp | exp  
exp → exp + term | exp - term | term  
term → term \* unario | term / unario | term % unario | unario  
unario → !unario | - unario | factor  
factor → ( bool ) | **id** localizacion | **numero** | **cadena** | **true**  
| **false** | **id**(parametros) | **id**  
parametros → lista\_param | ε  
lista\_param → lista\_param , bool | bool  
localizacion → localizacion ( bool ) | ( bool )