**Correcion gramática post-entrega de compilador**

Buenos días,

Corregido el trabajo práctico integrador la nota será definida luego del coloquio del día de hoy, igualmente esta muy bien y solo hay unas cositas a señalar:

La gramática de condiciones es incorrecta, permite realizar comparaciones con varios operadores lógicos y además permite realizar comparaciones incorrectas como (3>5)>(4>2) por ejemplo.

Error en la gramática de expresiones porque una constante string no puede ser un factor porque no permitimos operaciones aritméticas con constantes strings según la consigna del tp.

-------------------------------------------------------------------------------------------------------------------------------------------------------------

En color rojo se muestra el código viejo y erróneo, en color verde dejo las modificaciones para ver si son las correctas.

-------------------------------------------------------------------------------------------------------------------------------------------------------------

<programa> ::= <declaraciones> BEGIN\_PROGRAM <lista\_sentencias> END\_PROGRAM

| BEGIN\_PROGRAM <funcion\_print> END\_PROGRAM

<declaraciones> ::= DECLARE <lista\_declaraciones> ENDDECLARE

<lista\_declaraciones> ::= <lista\_declaraciones> <linea\_declaracion> | <linea\_declaracion>

<linea\_declaracion> ::= COR\_ABRE <declaracion> COR\_CIERRA

<declaracion> ::= ID COMA <declaracion> COMA <tipo> | ID COR\_CIERRA ASIGNA COR\_ABRE <tipo>

<tipo> ::= INT | FLOAT | STRING

<lista\_sentencias> ::= <lista\_sentencias> <sentencia> | <sentencia>

<sentencia> ::= <asignacion> | <funcion\_print> | <funcion\_if> | <funcion\_while>

<asignacion> ::= ID ASIGNA <expresion>

<expresion> ::= <expresion> SUMA <termino> | <expresion> RESTA <termino> | <termino>

<termino> ::= <termino> MULTIPLICA <factor> | <termino> DIVIDE <factor> | <factor>

-------------------------------------------ANTES----------------------------------------------

<factor> ::= ID | CTE\_INT | CTE\_FLOAT | CTE\_STRING  | PAR\_ABRE <expresion> PAR\_CIERRA | <funcion\_take>

------------------------------------------AHORA-----------------------------------------------

<factor> ::= ID | CTE\_INT | CTE\_FLOAT | PAR\_ABRE <expresion> PAR\_CIERRA | <funcion\_take>

Se elimina CTE\_STRING de la regla <factor>

------------------------------------------AHORA-----------------------------------------------

<funcion\_take> ::= TAKE PAR\_ABRE <op\_num> PUNTO\_COMA CTE\_INT PUNTO\_COMA COR\_ABRE <lista\_take> COR\_CIERRA PAR\_CIERRA

| TAKE PAR\_ABRE <op\_num> PUNTO\_COMA CTE\_INT PUNTO\_COMA COR\_ABRE COR\_CIERRA PAR\_CIERRA

<op\_sum> ::= SUMA | RESTA | DIVIDE | MULTIPLICA

<lista\_take> ::= <lista\_take> PUNTO\_COMA CTE\_INT | CTE\_INT

<funcion\_print> ::= PRINT CTE\_STRING

<funcion\_if> ::= IF PAR\_ABRE <lista\_condiciones> PAR\_CIERRA <bloque>

| IF PAR\_ABRE <lista\_condiciones> PAR\_CIERRA <bloque> ELSE <bloque>

<funcion\_while> ::= WHILE PAR\_ABRE <lista\_condiciones> PAR\_CIERRA <bloque>

-------------------------------------------ANTES----------------------------------------------

<condicion> ::= PAR\_ABRE <condicion> PAR\_CIERRA <op\_bin> PAR\_ABRE <condicion> PAR\_CIERRA

| PAR\_ABRE <condicion> PAR\_CIERRA <op\_logico> PAR\_ABRE <condicion> PAR\_CIERRA

           | PAR\_ABRE <condicion> PAR\_CIERRA <op\_logico> <expresion>

           | <expresion> <op\_logico> PAR\_ABRE <condicion> PAR\_CIERRA

           | <expresion> <op\_logico> <expresion>

------------------------------------------AHORA-----------------------------------------------

<lista\_condiciones> ::= PAR\_ABRE <condicion> PAR\_CIERRA op\_bin PAR\_ABRE <condicion> PAR\_CIERRA

| PAR\_ABRE <condicion> PAR\_CIERRA

<condicion> ::= <expresion> <op\_logico> <expresion>

Se modifica la manera en que se construyen las condiciones (se desglosa en dos reglas <lista\_condiciones> y <condicion>) y se editan las funciones que usan condiciones (IF y WHILE).

------------------------------------------AHORA-----------------------------------------------

<bloque> ::= LLAVE\_ABRE <lista\_sentencias> LLAVE\_CIERRA

<op\_logico> ::= IGUAL | DISTINTO | MENOR | MAYOR | MAYOR\_IGUAL

<op\_bin> ::= AND | OR