

Regla	Acciones semánticas
S → PROG	generarAssembler(_pProg, obtenerTS()); guardarTS(); generarArbolTXT(_pProg); generarGraphviz(_pProg);
PROG → SENT	_pProg = _pSent; crearNodosValidacion(funcionPosicion, funcionRead, existeLista);
PROG → PROG SENT	_pProg = newNode(";", _pProg, _pSent); crearNodosValidacion(funcionPosicion, funcionRead, existeLista);
SENT → READ	_pSent = _pRead;
SENT → WRITE	_pSent = _pWrite;
SENT → ASIG	_pSent = _pAsig;
READ → read id	insertarEnTS(bufferTS, puntBufferTs, \$2, TIPO_INT); _pRead = newNode("READ", NULL, newLeaf(\$2)); funcionPosicion = 0; funcionRead = 1;
ASIG → id asigna POSICION	insertarEnTS(bufferTS, puntBufferTs, \$1, TIPO_INT); _pAsig = newNode("=", newLeaf(\$1), _pPosicion); funcionPosicion = 1; funcionRead = 0;
POSICION → posición para id pyc ca	validarVariablePivot(\$3);
LISTA cc parc	existeLista = 1; aumentarContadorVariableResultado(); limpiarLista();
POSICION → posición para id pyc ca	validarVariablePivot(\$3);
cc parc	existeLista = 0; _pPosicion = newLeaf("_valorNoDeterminado");
LISTA → cte	indicePosicion = 0; insertarTS(\$1) insertarTS(insertarLista(\$1, indicePosicion)); insertarTS(obtenerStringVariableResultado()); _pCondPosicion = crearNodoVerificacionCoincidenciaPivot(); _pPosicion = newNode(";", crearNodoValorNoDeterminado(), _pCondPosicion);
LISTA → LISTA coma cte	indicePosicion++; insertarTS(\$3) insertarTS(insertarLista(\$3, indicePosicion)); _pCondPosicion = crearNodoVerificacionCoincidenciaPivot(); _pPosicion = newNode(";", _pPosicion, _pCondPosicion);
WRITE → write cte_s	insertarTS(\$2); _pWrite = newNode("WRITE", NULL, newLeaf(\$2)); funcionPosicion = 0; funcionRead = 0;
WRITE → write id	validarVariable(\$2); _pWrite = newNode("WRITE", NULL, newLeaf(\$2)); funcionPosicion = 0; funcionRead = 0;

### Funciones utilizadas:

- **crearNodosValidacion:** Verifica los valores de las flags funcionPosicion, funcionRead y existeLista para determinar si es necesario crear nodos de validación. En caso de tener funcionPosicion y existeLista, se creará el nodo para validar si no se encontró el elemento ingresado por consola. Por otra parte, si no se tiene lista, se deberá mostrar el mensaje “Lista vacía”. En cuanto a la flag funcionRead, se utiliza para insertar un nodo en el cual se valida si el pivot ingresado por el usuario es mayor o igual a 1.
- **insertarEnTS:** Función encargada de pasar los valores a los buffers temporales y realizar la inserción en la tabla de símbolos.
- **validarVariablePivot:** Pasa el nombre del pivot a un buffer temporal y comprueba si ya existe en la tabla de símbolos. En caso negativo, informa por pantalla y detiene la ejecución.

- aumentarContadorVariableResultado: Se concatena un número a la variable “@resultado”, de forma tal que en un mismo programa se pueda tener más de una sentencia con la función posición.
- limpiarLista: Una vez que ya se determinó el valor de resultado, debo limpiar la lista, de forma tal que, si se vuelve a llamar a la función posición, la misma no trae los valores del primer llamado de la función.
- insertarLista: Se pasa como parámetro la constante y la posición de la misma en el vector. La misma se utilizará para devolver correctamente el primer valor de coincidencia para un pivot.
- obtenerStringVariableResultado: Se obtiene el string “@resultado...” con el número actual.
- crearNodoVerificacionCoincidenciaPivot: Se crea un nodo de tipo IF, el cual tiene como condición que el pivot ingresado por el usuario coincida con el elemento actual. En caso verdadero, entonces se asignará el índice posición a la variable “@resultado...”.
- crearNodoValorNoDeterminado: Nodo en el cual se asigna el valor por defecto a la variable “@resultado...”. Dicho valor es utilizado luego para realizar validaciones como, por ejemplo, de elemento no encontrado.
- validarVariable: Antes de crear el nodo para write, se verifica que el id haya sido declarado.