Analizador Semántico

**Estructura del stack:**

El stack tendrá la estructura de una lista, y sólo contendrá los números que representen cada scope de la tabla de símbolos, además en la primera posición siempre contendrá el número 0 que representará la tabla de símbolos global y éste no podrá ser sacado del stack.

Un ejemplo del stack sería este:

* Stack = [0,2,3]

En donde significa que el scope 3 está dentro del scope 2, y a su vez éste está dentro del scope 0.

**Estructura de la tabla de símbolos:**

La tabla de símbolos correspondiente a cada scope, se encontrará dentro de un diccionario de diccionarios, siendo identificada cada una mediante un valor entero, que además representa el orden en el que fueron creadas.

Cada una de estas tablas tendrá la siguiente estructura:

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre del atributo | Tipo de variable | Notas | Ejemplo |
| Identificador del atributo | Cadena de caracteres | Con este se va a encontrar cada variable o función dentro de la tabla. | “main” |
| Tipo de variable o función | Cadena de caracteres | Este va a representar el tipo de la variable o función, pudiendo ser “void” o “int”. | “int” |
| Si es arreglo | Booleano | Este atributo será “True” en caso de que la variable que representa sea un arreglo y “False” en cualquier otro caso. | False |
| Tamaño del arreglo o número de parámetros de la función | Entero | Este atributo tendrá un valor de -1 cuando el identificador en cuestión no sea arreglo ni función.  En caso de ser función tendrá el número de parámetros correspondientes y en caso de ser arreglo tendrá el tamaño del arreglo. | 0 |
| Scope de la función | Entero | En caso de que el atributo en cuestión sea una función, aquí estará el scope correspondiente a esta.  En caso de que sea cualquier otro tipo de atributo, tendrá el valor de -1. | 5 |

**Reglas lógicas de inferencia de tipos:**

1. Identificadores:
   1. TablaDeSimbolos(Id) = INTEGER → ⊢ Id: INTEGER
2. Arreglos:
   1. TablaDeSimbolos(Id) = ARRAY → ⊢ Id: ARRAY
3. Identificadores:
   1. Identificador es una literal entera → ⊢ Identificador: INTEGER
4. Constantes:
   1. Constante → ⊢ Constante: INTEGER
5. Arreglos:
   1. Identificador es un arreglo → ⊢ Identificador: ARRAY
6. Operadores lógicos:
   1. String está dentro de los operadores lógicos → ⊢ String: BOOLEAN
7. Operadores enteros:
   1. String está dentro de los operadores → ⊢ String: INTEGER
8. Asignación:
   1. String es de tipo assign → ⊢ String: INTEGER
9. Constante dentro de los corchetes de un arreglo:
   1. ⊢ Identificador: ARRAY ^ ⊢ Constante: INTEGER → ⊢ Identificador [Constante]: INTEGER
10. Variable dentro de los corchetes de un arreglo:
    1. ⊢ Identificador1: ARRAY ^ ⊢ Identificador2: INTEGER → ⊢Identificador1[Identificador2]: INTEGER
11. Suma:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1+Identificador2: INTEGER
12. Resta:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1-Identificador2: INTEGER
13. División:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1/Identificador2: INTEGER
14. Multiplicación:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1\*Identificador2: INTEGER
15. Mayor que:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1>Identificador2: BOOLEAN
16. Menor que:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1<Identificador2: BOOLEAN
17. Menor o igual que:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1<=Identificador2: BOOLEAN
18. Mayor o igual que:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1>=Identificador2: BOOLEAN
19. Igual que:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1==Identificador2: BOOLEAN
20. Diferente que:
    1. ⊢ Id1: INTEGER ^ ⊢ Id2: INTEGER → ⊢Identificador1 !=Identificador2: BOOLEAN
21. Estatuto If:
    1. ⊢ OP: BOOLEAN → ⊢ if(OP):BOOLEAN
22. Estatuto while:
    1. ⊢ OP: BOOLEAN → ⊢ while(OP):BOOLEAN
23. Estatuto return:
    1. ⊢ EXP: INTEGER→ ⊢ return EXP:INTEGER
24. Estatuto call:
    1. ⊢TablaDeSimbolos(parametro) = INTEGER ^ ⊢TablaDeSimbolos(id) = INTEGER → ⊢
25. Estatuto call:
    1. ⊢TablaDeSimbolos(parametro) = INTEGER ^ ⊢TablaDeSimbolos(id) = INTEGER → ⊢Es correcto
26. Estatuto call:
    1. ⊢TablaDeSimbolos(parametro) = ARRAY ^ ⊢TablaDeSimbolos(id) = ARRAY → ⊢Es correcto