

Lenguaje PLO

GRAMÁTICA:

program = block "."

block = ["const" ident "=" number {"," ident "=" number} ";"]
["var" ident {"," ident} ";"]
{"procedure" ident "," block ";"} statement

statement = [ident ":"= expression | "call" ident
| "?" ident | "!" expression
| "begin" statement {"," statement} "end"
| "if" condition "then" statement
| "while" condition "do" statement]

condition = "odd" expression |
expression ("=" | "<" | "<=" | ">" | ">=") expression

expression = ["+" | "-"] term { ("+" | "-") term }

term = factor { ("*" | "/") factor }

factor = ident | number | "(" expression ")"

EJEMPLO:

```
VAR x, squ;  
  
PROCEDURE square;  
BEGIN  
    squ:= x * x  
END;  
  
BEGIN  
    x := 1;  
    WHILE x <= 10 DO  
        BEGIN  
            CALL square;  
            ! squ;  
            x := x + 1  
        END  
    END  
END.
```

TABLAS

TABLA DE SÍMBOLOS

Símbolo: Cualquier elemento con nombre que el programador haya establecido como válido dentro de él a través de una declaración.

La tabla es dinámica dada la naturaleza de los elementos dentro de ella, ya que son definidos por el usuario

Notas: ¿Llevará niveles?

TABLA DE TIPOS

TIPO: Descriptor semántico que permite cualificar al símbolo al que está vinculado de manera que condiciona su interpretación semántica y restringe las operaciones que se pueden realizar sobre él.



Trabajando en ello

Ejemplo: (Tipos primitivos)

| Tipo | Descripción | Rango | Expresión | Operaciones |
|---------|--|--------------------|-----------|----------------------|
| Byte | Representación numérica corta sin signo. | 0 a 255 | Entero | + - * / mod = > < <> |
| Integer | Representación numérica con signo. | -32768 a 32767 | Entero | + - * / mod = > < <> |
| Word | Representación numérica larga sin signo. | 0 a 65535 | Entero | + - * / mod = > < <> |
| Real | Representación flotante corta con signo. | 2.9E-39 a 1.7E308 | Real | + - * / = > < <> |
| Double | Representación flotante larga con signo. | 5.0E-324 a 1.7E308 | Real | + - * / = > < <> |
| Boolean | Representación lógica. | False, True | Lógico | AND OR XOR NOT |
| Char | Representación de carácter. | ASCII | Carácter | - |
| String | Representación de cadena. | - | Cadena | - |

TABLA DE SIGNOS

- Identificar todos los operadores utilizados en el lenguaje.
- Nombre de cada uno
- Identificar el bloque de código donde se ocupa

| Operador | Significado | Utilización |
|----------|-------------------|--|
| . | Punto | Fin del programa |
| = | Igual | Asignación de constantes Comparación en condicional |
| ; | Punto y coma | Fin de la declaración |
| := | Dos puntos igual | Asignación de variables |
| ? | | |
| ! | | |
| # | | |
| > | Mayor que | Comparaciones |
| >= | Mayor o igual | |
| < | Menor que | |
| <= | Menor o igual | |
| + | Suma | Operaciones aritméticas |
| - | Resta | |
| * | Multiplicación | |
| / | División | |
| (| Paréntesis abre | |
|) | Paréntesis cierra | |

*Se encuentra en sombreado azul lo que posiblemente se eliminará

TABLAS DE VERIFICACIÓN DE TIPOS

Se hace una tabla de verificación de tipos por cada signo del lenguaje.

TABLA DE PALABRAS RESERVADAS

| Palabra | Se refiere a: |
|-----------|-------------------------------|
| IF | |
| DO | |
| VAR | Variable |
| END | Finaliza el bloque de código. |
| ODD | |
| CALL | Llamada a función. |
| THEN | |
| BEGIN | Inicia el bloque de código. |
| WHILE | |
| CONST | Constante. |
| PROCEDURE | Procedimiento. |