

Compiladores: Lenguaje de programación

Alumno: Sebastian Wilde Alarcón Arenas

Tipos de datos	
int	ent
float	dec
bool	bin
char	car
string	cad
void	vac

Condicionales	
Y	&
O	
Mayor y menos	> / <
Mayor igual, menor igual	>= / <=
Igual	==
diferente	/=
Operaciones	
Suma	+
Resta	-
Multipliación	*
División	/

Otros	
main	ini
bucle	bucle
return	regresar
function	def
Variables	var
Llamar funcion	llamar
Break	cortar
If	Si
Else if	quizas
Else	sino

Consideraciones del lenguaje:

- Los indicadores de inicio o fin de un ámbito son “{}” (llaves)
- El indicador de fin de una sentencia es “.” (punto)

Sobre el inicio del programa:

Para iniciar el programa se debe comenzar con la palabra reservada “ini” seguido de un signo de llave abierto “{” seguidamente se colocará la palabra reservada “var” seguido del símbolo de llaves abierta “{” es aquí donde se declararán todas las variables y al terminar el símbolo de llaves “}”, después todo el cuerpo del programa y terminará con un signo de llave cerrado “}”. Siempre debe acabar con retornar 0.

Ejemplo:

```
Ent ini
{
    Var
    {
        Declaracion de variables
    }
    Cuerpo
    retornar 0.
}
```

Sobre las operaciones:

Para una operación se necesita mínimo 2 componentes, entre ellos el operador correspondiente. No está permitido operaciones entre diferentes tipos de variables. Se puede tener operaciones en cadena pero estos deben estar agrupados entre paréntesis “()” y se termina con el “.”

Ejemplo:

$a + b. / 5 * 6. / 3 + (a + ((b/c)+c) - c .$

Sobre las condicionales:

Para una operación se necesita mínimo 2 componentes, entre ellos el operador condicional correspondiente. No está permitido operaciones entre diferentes tipos de variables. Se puede tener operaciones en cadena pero estos deben estar agrupados entre paréntesis “()” y se termina con el “.” Se inicia con un “si” seguidamente la condición entre paréntesis “()” y entre llaves el código “{}”, para hacer un elseif se usa la palabra reservada “quizas” ,seguidamente la condición entre paréntesis “()” y entre llaves el código “{}”. Y para hacer un else se usa la palabra reservada “sino” seguidamente entre llaves el código “{}”

Ejemplo:

$a \& c. / 0 \mid 1. / (a == c) \& (a/=2) .$

```

si (a<5)
{
    Retornar 7.
}
quizas (a = 3)
{
    retornar 0.
}
sino
{
    cortar.
}

```

Sobre las declaraciones de variables:

Para declarar una variable primero debe estar el tipo de variable (entero: ent, decimal: des, binario: bin, carácter: car, cadena: cad, vacio: vac) seguidamente del nombre de la variable (solo está permitido letras minúsculas) y números, las variables no pueden comenzar con números. Seguidamente puede terminar la sentencia con un “.” o asignar el valor correspondiente con el símbolo “=”, el valor y el “.” para terminar la sentencia.

Ejemplos:**Válido**

```

var
{
    ent a.
    dec b2.
    bit aux = 0.
}

```

No válido

ent 2abc

Sobre la declaración de funciones:

Para declarar una función se inicia con la palabra reservada “def” seguido del tipo de dato que retornará la función, seguidamente del nombre de la función (solo está permitido letras minúsculas) y números, las variables no pueden comenzar con números. Después los parámetros que estarán (máximo 5 parámetros) entre símbolos de paréntesis “()”, si no hubiera ningún parámetro deja en blanco, caso contrario cada parámetro debe tener el tipo de dato, el nombre del parámetro y seguido de una coma “,” en caso de que haya más. Después un espacio si se declaran variables se colocará la palabra reservada “var” seguido del símbolo de llaves abiertas “{” es aquí donde se declararán todas las variables y al terminar el símbolo de llaves “}” después todo el cuerpo de la función y en la última línea de la función va la palabra reservada “regresar” y el nombre de la variable que va a retornar o el valor seguido del “.”. Para llamar una función dentro de un programa se colocará la palabra reservada “llamar”, el nombre de la función, los parámetros entre paréntesis “()” y el símbolo “.” para indicar que terminó la sentencia.

Ejemplo:

```
def ent fun1 (ent abc, float bca)
{
    Var
    {
        ent b.
    }
    Cuerpo
    retornar bca.
}
```

Llamar a una función: llamar nombreFuncion (parámetros).

Sobre los bucles

Para iniciar un bucle se debe iniciar con la palabra reservada “bucle” seguido de la condición entre paréntesis “()” y seguido del cuerpo del bucle entre llaves “{}”. Al igual que los otros ejemplos si se declararán variables debe hacerse como está especificado en las secciones previas. Se puede usar “cortar” para terminar con el bucle.

Ejemplo del bucle:

```
bucle (a<5)
{
    Var
    {
        ent b.
        cortar.
    }
    cuerpo
}
```