

Especificación del Lenguaje

Johan Gonzalez

Contents

Introducción	2
Estructura Léxica	2
Espacios en Blanco	2
Comentarios	2
Identificadores	2
Palabras reservadas	2
Tipos	2
Literales	2
Literales Numéricos (enteros)	2
Literales Booleanos	3
Operadores	3
Estructura del programa	3
Declaracion de variables	3
Declaraicion de Funciones y procedimiento	3
Argumentos de Funciones y procedimientos	3
Asignación	4
CondicionaI If	4
Impresion de valor	4
LLamada a procedimiento	4
Argumentos de llamada	4
Retorno	4
Ejemplo	5

Introducción

En este documento se especifica el lenguaje DymScope. Es un lenguaje que tiene como objetivo estudiar alcances y asociaciones, por lo que debe poder ejecutarse con alcance dinámico y estático, y asociaciones profundas y superficiales.

Estructura Léxica

Espacios en Blanco

Los espacios en blanco están definidos por los caracteres ASCII de espacio, tabulación, salto de línea.

Comentarios

Existen dos tipos de comentarios:

- `/texto/` – donde todo el texto entre `/*` y `*/` es ignorado.
- `//texto` – donde todo el texto desde `//` hasta el final de la línea es ignorado.

Identificadores

Un identificador es una secuencia ilimitada de caracteres alfanuméricos, la cual debe comenzar con una letra.

Un identificador no puede ser igual a alguna palabra reservada o constante booleana.

Palabras reservadas

Palabras del lenguaje que no pueden ser redefinidas ni usadas como identificador:

- **function**
- **return**
- **proc**
- **if**
- **else**
- **var**
- **int**
- **print**

Tipos

Existen tres tipos básicos:

- **int**
- **function**
- **proc**

Literales

Literales Numéricos (enteros)

Los literales enteros pueden ser expresados en decimal(base 10) y hexadecimal(base 16)

Literales Booleanos

Las dos constantes booleanas:

- `true`
- `false`

Operadores

```
+ - / * = < > <= >= != == && || !
```

Estructura del programa

Un programa esta definido como una serie de instrucciones y/o declaraciones, las cuales son ejecutadas secuencialmente. Las posibles instrucciones y declaraciones posibles son:

- Declaracion de variable
- Declaracion de función
- Declaracion de procedimiento
- Asignacion
- Condicional if
- LLamada a procedimiento
- Retorno
- Impresion de Valor

Declaracion de variables

Introduce un identificador para referirse a un valor, este valor puede ser obtenido y modificado por su nombre a lo largo de su alcance.

```
var identificador : tipo ;
```

Declaraicion de Funciones y procedimiento

Introduce una serie de instrucciones que pueden ser invocados a través de su nombre y los argumentos requeridos por la función.

```
function identificador ( argumentos ) { instrucciones }  
proc identificador ( argumentos ) { instrucciones }
```

Argumentos de Funciones y procedimientos

Es una lista separada por ‘,’ ,posiblemente vacía de identificadores seguidos por su tipo.

```
identificador : tipo , argumento  
identificador : tipo  
''
```

Asignación

Asigna un nuevo valor a la variable.

```
variable = expresión;
```

Condicional If

Se evalúa la expresión booleana cond, y en caso de evaluar true, se ejecuta el bloque trueBlock, de lo contrario se ejecuta el bloque false en caso de existir.

```
if (cond) { trueblock } else { falseblock }  
if (cond) { trueblock }
```

Impresion de valor

Imprime en pantalla el valor pasado como argumento

```
print ( expresión ) ;
```

LLamada a procedimiento

Ejecuta el procedimiento identificador, con los valores de los argumentos. Los argumentos son evaluados de izquierda a derecha.

```
identificador ( argumentos ) ;
```

Argumentos de llamada

Es una lista separada por ‘,’ de expresiones, que puede estar vacía.

```
expresión , expresión  
expresión  
''
```

Retorno

Culmina la ejecuciones de la función o procedimiento dando como resultado la expresión especificada. La expresión es opcional

```
return ;  
return expresión ;
```

Ejemplo

```
var n:int;
n=10;

function fib(n:int){
    if(n==0){
        return 1;
    }
    if(n==1){
        return 1;
    }
    return fib(n-1) + fib(n-2);
}

print(n);
print(fib(n));
```