Universidad Simón Bolívar

Traductores e Interpretadores

Informe del Analizador Lexicográfico para el lenguaje BasicTran

Autores: Irina Marcano 13-10805 :: Fernando González 08-10464

**Estructura de la entrega:**

Se crea una clase llamada **ElementoTexto** la cual presenta como atributos línea, columna y contenido. A partir de la premisa de que los elementos a representar del lenguaje son “objetos de tipo texto” se crean dos clases centrales que son **Token** y **Error**, ambas clases heredan sus atributos de objeto de texto. Dichos atributos contendrán la información pertinente para llevar a cabo la impresión de la información deseada.

Seguido a esto pasamos a definir el diccionario de las RegExp para los tokens definidos en el enunciado del lenguaje. Cada elemento del diccionario está definido de la forma:

**... , <Token\_name> => <RegExp> , …**

Donde **Token\_name** puede tomar los valores de:

**Punto || Num || Caracter || Id || Coma || DosPuntos || ParAbre || ParCierra || CorcheteAbre || CorcheteCierre || LlaveAbre || LlaveCierra || Resta || Hacer || Asignacion || Suma || Desigualdad || PuntoYComa ||Mult || Div || Mod || Conjuncion || Disyuncion || Negacion || Menor || MenorIgual ||Mayor || MayorIgual || Igual || SiguienteCar || AnteriorCar || ValorAscii || Concatenacion || Shift**

y **RegExp** sus respectivas expresiones regulares.

Escribimos las palabras reservadas para nuestro lenguaje:

**( with , true , false , var , begin , end , int , while , if , else , bool , char , array , read , of )**

Y de igual manera las introducimos en el diccionario antes definido para la correcta identificación de nuestros Tokens.

Y para cada uno de nuestros tipos de tokens definidos en el diccionario se crea una clase, dichas clases pertenecen a la clase Token y se inicializan para tener el contexto del token que serían **Línea**, **Columna** de su posición y su posible **contenido** y procedemos a renombrar cada clase anteponiendo la abreviatura Tk a las palabras de nuestro diccionario.