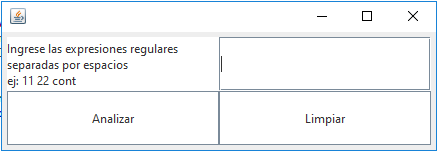
Universidad Distrital Francisco José de Caldas Facultad de Ingeniería Proyecto Curricular Ingeniería de Sistemas Ciencias de la Computación 3 Bogotá 2019

Sebastián Mora Sabogal – 20161020031

1. ENUNCIADO:

Realizar un analizador léxico que distinga entre números enteros, números reales e identificadores de usuario.

1.2 INTERFAZ GRÁFICA DE USUARIO



* + 1. Componentes de la Interfaz.
* tfExpresiones: Campo de texto donde se ingresan las expresiones a ser analizadas, las expresiones deben ser escritas como se muestra en el JLabel ubicado a la izquierda. Este componente se encuentra ubicado en la parte superior derecha de la interfaz.
* btnAnalizar: El botón con el texto “Analizar”, el cual tiene como función llamar a la lógica para que esta se encargue de analizar las entradas ubicadas en tfExpresiones. Se encuentra ubicado en la parte inferior derecha.
* btnLimpiar: El botón con el texto “Limpiar”, se encarga de dejar el campo tfExpresiones sin texto para que se vuelva a llenar con otras expresiones de entrada. Se encuentra ubicado en la parte inferior derecha.
  1. LÓGICA

Se encarga de realizar el análisis sintáctico de la entrada que obtendrá como argumento. A continuación el código fuente de la clase:

**public** **class** Logica {

**public** String Analizar(String expresiones) {

**char**[] caracteres = expresiones.toCharArray();

**boolean** isinteger = **false**;

**boolean** isreal = **false**;

**boolean** isID = **false**;

String resultado = "";

**for** (**int** i = 0; i < caracteres.length; i++) {

**if**(Character.*isLetter*(caracteres[i]) || caracteres[i]=='\_' || caracteres[i]=='&') {

isinteger = **false**;

isreal = **false**;

isID = **true**;

}

**if**(isinteger == **true** && (caracteres[i] == '.' || caracteres[i] == ',')) {

isreal = **true**;

isinteger = **false**;

isID = **false**;

}

**if** (Character.*isDigit*(caracteres[i]) && isreal == **true** ) {

isreal = **true**;

isinteger = **false**;

isID = **false**;

}**else** **if**(Character.*isDigit*(caracteres[i])) {

isID = **false**;

isinteger = **true**;

isreal = **false**;

}

**if**(caracteres[i] == ' ' || i == caracteres.length -1 ) {

**if**(isinteger == **true** && isreal == **false** && isID == **false**) {

resultado += "NUM\_ENT ";

}

**if**(isreal == **true** && isinteger == **false** && isID == **false**) {

resultado += "NUM\_REAL ";

}

**if** (isID == **true** && isreal == **false** && isinteger == **false**) {

resultado += "ID ";

}

isinteger = **false**;

isreal = **false**;

isID = **false**;

}

}

**return** resultado;

}

}

Una descripción básica de lo que hace la función es descomponer la entrada que tiene las diferentes expresiones convirtiéndola en una lista de caracteres, esto con la intención de analizar los caracteres de una forma similar como lo haría un analizador léxico para comparar las expresiones con los patrones correspondientes.