



Taller de Programación de Sistemas CC207

NRC 07073

MARIA ELENA ROMERO GASTELU

2013-B Sección D02

Práctica #1

Alan Andrés Sánchez Castro 208697345

Descripción de Clases, Variables y Métodos utilizados

En la primera práctica creé las clases

- Codop
- Etiqueta
- Operando
- Linea
- Util
- Gui

Gui

En esta clase se crea la interfaz gráfica de la aplicación (usé tkinter para ese fin).

El único método que no es del todo para la interfaz es *cargarArchivo*, en donde se le pide al usuario un archivo de texto, lo abre y pone su contenido en un campo de texto. Si el usuario usa un archivo válido, empieza a parsearlo línea por línea. Cada una de estas líneas es mandada a instancias de la clase *Linea*. Si se van encontrando errores en cada *Linea*, se irán mostrando en el panel de texto dedicado para ese fin (Erro Log) y los estados de las variables serán mostrados en el panel Salida. Además, el método evalúa si el archivo contiene el ***codop END***. Para esto se utiliza una bandera: *banderaEnd*, que es local al método antes mencionado.

Linea

En esta clase se evalúa una línea en código ensamblador.

Contiene las variables

- etiqueta :*Etiqueta*
- codop :*Codop*
- operando :*Operando*
- comentario :*Bool*
- errores :*List()*

Además contiene los métodos

- *atributos :String*
- *getErrores :List()*

Las variables etiqueta, codop y operando es donde se guardarán ya aislados estos códigos. La variable comentario indica si la línea que se está evaluando es comentario o no. La variable errores es donde se guardarán los errores que vayan generando *Etiqueta*, *Codop* y *Operando*, en caso de que estos existan, además de los posibles errores que puede contener la línea por sí sola (por ejemplo que sea una línea vacía).

El algoritmo utilizado para evaluar la línea es el siguiente:

Primero se evalúa si el primer carácter es ‘;’, si lo es entonces se sabe que es un comentario y se termina de evaluar la línea. En caso contrario, primero se hace una copia de la línea en una variable auxiliar, y a la variable con el texto original se le rempazan los ‘\t’ por ‘ ’, para después pasar a subdividirla en bloques separados por espacios en blanco.

Ya que se tiene la línea separada por bloques, se procede a evaluar el primer carácter, si es un espacio en blanco, entonces se sabe que no tendrá Etiqueta, por lo tanto el primer bloque será el Codop, y en caso de existir dos o más bloques, a partir del segundo serán los que conformen el Operando. En caso contrario, si el primer carácter de la línea no es un espacio en blanco, entonces el primer bloque será la Etiqueta, el segundo (si es que existe) será el Codop, y si existen tres o más bloques, a partir del tercero conformarán el Operando.

Para sacar el Operando original (sin cambiar los ‘\t’ por ‘ ’), se utiliza el texto original de la línea que fue almacenado en la variable auxiliar.

Cabe mencionar que a las clases *Etiqueta*, *Codop* y *Operando* se le pasan como argumento los respectivos bloques de la línea que fueron obtenidos anteriormente.

Después de que se tiene (si es que existen) la Etiqueta, Codop y Operando, además de sus respectivos errores guardados en la lista *errores*, se procede a evaluar si la línea por sí sola contiene más errores (por ejemplo que el Codop no exista) y en caso de tenerlos, éstos se almacenan.

Codop, Etiqueta y Operando

Reciben como argumento una cadena de texto y una lista de errores, donde se irán agregando los errores encontrados.

Sus variables son

- `text :String`
- `errorList :List()`
- `errores :List()`

Y sus métodos

- `validar :Void`
- `getErrores :List()`

El método ***validar*** es el que hace el trabajo de evaluar la cadena. Aquí se evalúa el tamaño de la cadena y después, carácter por carácter, para saber los errores que puede o no contener.

En el caso de *Etiqueta* se utiliza una bandera para saber si la cadena contiene más de un ‘.’.

Util

Sólo contiene dos métodos

- existeArchivo
- stringIsNullOrEmpty

que son utilizados para saber si un archivo existe y si una cadena está vacía o es Null.

En ésta clase sólo se encuentran dos métodos

Algoritmo utilizado para la evaluación de las líneas

El algoritmo utilizado es separar el archivo en cada una de sus líneas.

Cada línea a su vez es separada por espacios en blanco o tabuladores, generando así una lista de bloques de texto.

Después cada bloque de texto es evaluado carácter por carácter.

Por lo tanto se puede decir que se utiliza un algoritmo secuencial.