

Informe de TP 3 Interprete SCEQL

75.40 Algoritmos y Programación I
Cátedra Wachenchauzer

Cuatrimestre II, 2017

Dificultades a afrontar

Implementar un objeto interpretador cuyos atributos y métodos nos ayuden a interpretar el lenguaje SCEQL. Entre los métodos se encuentran:

1. Una función que lea el archivo ingresado y valide los siguientes puntos:
 - a. Que el archivo pueda abrirse y/o tenga permisos (que no haya IO error)
 - b. Analice el contenido del archivo y que lo 'limpie' dejándole sólo los comandos del lenguaje SCEQL.
 - c. Que la cantidad de barras y contrabarras sean iguales.
2. Una función que lea que contenido "limpio" y modifique la cola y la traducción según cada caracter que va leyendo:
 - a. Esto implica la creación de un objeto Cola el cual mediante sus metodos pueda incrementar o decrementar su primer elemento sin modificar el orden de la cola, priorizando la optimización (ColaEnlazada).
3. Una función debug que hara lo mismo que la anterior pero además muestre el estado de la ejecución, es decir, la posición en el archivo SCEQL, la cola y la traducción

Resolución

1. Obtenemos el nombre del archivo que es ingresado por el usuario. Creamos el objeto "Interprete", en el cual dos métodos se encargaran de manipular el archivo ingresado. Vamos a recorrer el archivo una vez a traves del metodo **leer_archivo** y lo vamos a ir guardando en el atributo **self.contenido**, solo con los caracteres del lenguaje SCEQL, además si encontramos una barra, vamos a crear dos diccionarios, **self.barras_abiertas** y **self.barras_cerradas** con las posiciones de las barras como claves y de las otras barras como valores, de la siguiente forma $\text{dic}[X] = Y$ y viceversa $\text{dic}[Y] = X$. Esto nos ayudará posteriormente para saber donde debe desplazarse el intérprete cuando se encuentre con las barras ``` y `'`.
2. Creamos también el objeto "Cola Enlazada", que tiene los mismos métodos que una Cola pero como la hicimos en base a Nodo, y el constructor tiene un referencia al primero y al ultimo, para incrementar o decrementar, esto se puede resolver bastante más rápido. La función la llamamos **interpretar_valores**, que se desplazara por la cadena **self.contenido** según el **self.pos_act** que es la posición en la cadena al cual el la función se movera según corresponda. Y ademas, guardara cualquier output de caracter ASCII en el atributo **self.traduccion**.
3. En el modo debug, el programa pedira al usuario oprimir la tecla enter, esto lo logramos con un **while** y dentro del mismo se imprimira **self.traduccion**, **self.contenido** y un puntero que indicara cual es la **pos_act**.

