

# Proyecto 1

## JCF utilizados

### List

Se utilizó la estructura de List para poder guardar todas las instrucciones del texto y poder encajonarlas como aparece en el String del lenguaje LISP. Principalmente se escogió, ya que es la estructura principal en la cual se basa el LISP (List Processing). Además de aprovechó que el máximo tiempo de búsqueda sería  $O(n)$ , no el mejor, pero mucho mejor que otras estructuras.

### Stack

Se utilizó la estructura de Stack para trabajar los resultados de las operaciones y poder hacer la iteración de todos los parámetros de una operación dada. Se aprovechó que esta estructura tiene  $O(1)$  en complejidad para agregar y eliminar elementos, lo que hace nuestro algoritmo mucho más eficiente.

### HashMap

Se utilizó la estructura de un HASHMAP para poder crear un tipo diccionario para las variables y que sea mucho más fácil buscar sus valores asignados globalmente. Esto nos lo permite el hecho que el HashMap tiende a  $O(1)$  si la función de Hash está bien definida y produce pocas colisiones.