

Ayleen Rubio 19003

Daniela Batz 19214

Andrés Say 19705

Interpreté de Lisp

El Proyecto consistió en realizar un interprete de LISP para un subconjunto de instrucciones principales tales como; operaciones aritméticas, instrucciones QUOTE, funciones, predicados etc. Esto debía de realizarse a través de java con el objetivo de poder ser capaces de determinar el espacio de implementación y escoger correctamente la estructura de datos, además de poder identificar los elementos fundamentales y usos de la programación funcional.

Para llevar a cabo el interprete fue necesario utilizar un grupo de colección de funciones propia de java para poder organizar y evaluar los datos ingresados al programa, por lo tanto, se hizo uso de Java collection FrameWork

Java collection FrameWork

La plataforma Java incluye un marco de colecciones. Una colección es un objeto la cual representa un grupo de objetos. La interface más general es Iterable, la cual define que una colección que se puede recorrer devolviendo para este un Iterator. Ya que se tiene esta interface aparece la interface Collection que se define como el concepto de abstracto de una "Colección" a la que hace referencia a un conjunto de elementos que se pueden recorrer.

Las colecciones que usamos fueron las siguientes:

ArrayList: ArrayList forma parte de las colecciones de java e implementa la interfaz lista, un arrayList es una lista o array dinámico, puede crecer para acomodar los elementos que se ingresan o disminuir cuando los elementos son retirados. Se utilizó esta colección ya que permitía manejar los datos ingresados de una forma más eficaz, de esta manera era posible evaluar el String y verificar que contenía, si era una palabra reservada (como DEFUN), un numero o un operador, además se complementa con la otra colección que se utilizó para el manejo de datos, esta es la colección de lista. (Naftalin, 2007)

List: List es una interfaz de la colección de funciones de java, es una colección ordenada de objetos en donde se pueden almacenar varios elementos. Se utilizó list debido a que es posible manejarlo haciendo uso de la colección de arrayList, además la característica fundamental de List es que preserva el orden de inserción y permite el acceso por posición a estos elementos. (Álvarez, 2015)

HashMap:

Forma parte de la colección de funciones de java, permite utilizar la implementación de la interfaz Map para usar diccionarios, guarda una llave y su respectivo valor en pares, de esta manera nos podemos referir a un valor utilizando solamente su llave, además si tengo una llave es posible evaluarla en el diccionario hasta encontrar su respectivo valor (Naftalin, 2007). Los mapas se utilizaron para manejar las funciones y palabras reservadas de LISP, en donde los objetos están dentro del mapa, siendo la llave el nombre de la función. De esta manera cuando el lector obtenga una palabra reservada (véase DEFUN) hace un nuevo objeto y lo ingresa al mapa, esto se hace con el propósito de que al momento de llamar la función en el mapa este operará mediante algún método que tenga el objeto.

Ayleen Rubio 19003

Daniela Batz 19214

Andrés Say 19705

Referencias

- Naftalin M. Wadler P.(2007)9 JAVA Generics and collections. [Versión: electronica] extraído de:
https://books.google.com.gt/books?id=VUSbAgAAQBAJ&dq=list+in+java+collection+framework&hl=es&source=gbs_navlinks_s
- Álvarez, C., (2015). Java Collections List vs Set (I). Extraído de
<https://www.arquitecturajava.com/java-collections-list-vs-set/>