



UNIVERSIDAD
DE SANTIAGO
DE CHILE

Informe de Laboratorio

Emilio Viñals
Paradigmas de Programación
Gonzalo Martinez
23 Abril 2023

Introducción

En el siguiente informe se presentará y explicará el desarrollo de un programa, realizado en el lenguaje de programación Scheme, utilizando DrRacket, esto con la intención de aplicar lo aprendido en clases, el paradigma funcional.

El fin de esto es realizar una simulación de un sistema de archivos que usan los computadores, en base a listas inmutables, obteniendo y modificando la información actual que se quiera mostrar sin modificar ningún tipo de variables del entorno de trabajo.

Descripción del problema

El laboratorio consta de realizar una simulación de un sistema de archivos, esto contemplando el sistema operativo, usuarios, tipos de almacenamiento, carpetas, etcétera.

Para esto se usará el lenguaje de programación Scheme en DrRacket, como paradigma funcional, este tipo de programación se caracteriza por no usar variables. Al no existir variables ni nada que sea mutable, esto hace que las funciones sean específicas y que al momento de querer modificar algo, simplemente se crea una copia de lo que se quiere modificar, con los cambios necesarios.

Para poder realizar un programa como este, se ha diseñado un sistema a base de listas, el cual guarda la información completa en una lista, esto corresponde a, el nombre del sistema, letra del drive, nombre, almacenamiento, carpetas y archivos.

Análisis del problema - diseño de la solución

El problema consta en el desarrollar este sistema sin el uso de variables, en este caso con Scheme, para eso, es necesario crear un sistema a base de listas, con esto, se van creando nuevas listas cada vez que se quiera realizar algún tipo de modificación, como por ejemplo, crear un nuevo usuario, o una nueva carpeta, etc.

Para diseñar la solución, se hicieron pruebas en la cmd en un computador con windows, ya que las funciones exigidas en el laboratorio son las mismas que se pueden ejecutar por líneas de comandos en el cmd.

Al analizar el funcionamiento de estas líneas de comando en la cmd, se llegó a la conclusión, de que la manera más óptima de desarrollar el laboratorio es con una creación de listas, ya que al preguntar cosas como la carpeta actual o ir moviéndose entre carpetas, el cmd entrega un string con la posición en la que el usuario se encuentra, entonces se trabajará de esta forma para el desarrollo.

De esta forma, para poder llevarlo a Scheme se desarrollaron funciones que retornan una lista, esta lista contiene toda la información que uno quiera saber, entonces, al momento de querer cambiar de carpeta, usuario o disco etc, se retorna una nueva lista con la información solicitada.

Como ya se había mencionado anteriormente, el lenguaje de programación utilizado fue Scheme y su compilador fue DrRacket, las bibliotecas usadas fueron las por defecto del programa, no se agregó ninguna extra, esto debido a las indicaciones del laboratorio.

Instrucciones de uso

Para poder usar el programa desarrollado, se crean funciones simples, las cuales van llamando a otras en cadena para poder obtener resultados, como por ejemplo las siguientes:

Función para crear el sistema

```
(define S0 (system "Original"))  
S0
```

Función que usa el sistema anterior para añadirle un drive llamado S0 con una capacidad de 256gb

```
(define S1 ((run S0 add-drive) #\C "S0" 256))  
S1
```

De esta forma, el sistema se va enlazando continuamente para obtener los resultados deseados.

Resultados

Los resultados no fueron los esperados, ya que se esperaba poder completar el sistema.

Dentro de las funciones realizadas, se obtuvieron buenos resultados, todo esto gracias a la ayuda del profesor en clases.

Dentro de las funciones faltantes en el trabajo, se hicieron pruebas las cuales no llegaron a un resultado favorable, ya que constantemente terminaban en error, no había ninguna prueba que saliera como se esperaba, por lo cuál se optó por sacar del

programa, ya que al estar mal implementada generaba errores en el mismo código, lo que imposibilitaba el funcionamiento de las demás.

conclusiones.

En cuanto al desarrollo en sí, fue bastante complicado el cambiar el tipo de pensamiento al no tener variables las cuales modificar, el paradigma en sí fue parte de la complejidad, aunque en ciertos casos también fue de ayuda.

Al no poder usar variables al momento de desarrollar el proyecto, esto dificulta el desarrollo debido a la forma de pensar con la que llegó uno por los lenguajes de programación anteriormente usados en otras asignaturas.