

Paradigmas de la Programación

Informe Laboratorio nro. 2

Paradigma Lógico

*Profesor: Gonzalo Martínez.*

*Alumna: Karina Bustamante H.*

3 de junio de 2023

Tabla de contenido

[Introducción 2](#_Toc134312314)

[Descripción del problema 3](#_Toc134312315)

[Descripción del paradigma 3](#_Toc134312316)

[Análisis del problema 4](#_Toc134312317)

[Diseño de la solución 4](#_Toc134312318)

[Consideraciones de implementación 4](#_Toc134312319)

[Instrucciones de uso 5](#_Toc134312320)

[Resultados y evaluación 5](#_Toc134312321)

[Conclusiones 6](#_Toc134312322)

[Referencias 6](#_Toc134312323)

# Introducción

Para el desarrollo del laboratorio Nº2, de la asignatura de Paradigmas de la Programación, se nos solicita realizar una implementación bajo el paradigma lógico, el cual se basa en la lógica matemática y en la resolución de problemas mediante la deducción lógica. Los algoritmos a implementar deberán resolver el problema que corresponde al siguiente enunciado:

***“Crear un simulador de sistema operativo centrado específicamente en un sistema de archivos simplificado y los comandos que*** ***permiten operar sobre éste.”***

Para el desarrollo de esta implementación usaremos el lenguaje de programación, Prolog, con su editor virtual, (https://swish.swi-prolog.org/) y a través de la creación de una base de conocimientos que crearemos con predicados, reglas y clausulas, se buscará dar solución a los requerimientos funcionales solicitados, de manera tal, que la solución propuesta permita al usuario aplicar sobre este sistema de archivos ficticio, acciones como crear, buscar, listar, etc.

Desarrollando para cada acción, el algoritmo en

# Descripción del paradigma

El paradigma lógico corresponde al paradigma de programación declarativo, puede usarse como la especificación de un problema en lugar de establecer los pasos necesarios para llegar a una solución. Se basa en la lógica booleana y no existen funciones ni retornos normales. En este paradigma, se describe el problema en términos de relaciones lógicas entre objetos y se utiliza la inferencia lógica para resolver el problema. Se declara una base de conocimiento que es un conjunto de hechos y reglas que describen las relaciones entre los objetos en un dominio de problema. El programador describe el objetivo o la pregunta que se quiere responder en términos de estas relaciones lógicas. Luego, el sistema de programación utiliza la lógica para deducir la respuesta al objetivo a partir de los hechos y reglas definidos en la base de conocimiento. Los hechos, son un tipo de cláusula que describe una relación entre uno o más término, estos hechos se asumen siempre verdaderos.

Los mecanismos básicos de este paradigma son:

* ***Unificación***: se refiere a la técnica de encontrar una asignación de valores a las variables en una expresión lógica que hace que la expresión sea verdadera. Si se encuentra una asignación de valores que haga que las expresiones sean iguales, la consulta se considera resuelta y se devuelve el resultado
* ***Backtraking automático***: Sucede cuando no se logra la unificación, el proceso da un paso atrás para probar otros caminos para lograr la unificación, si se encuentra una asignación de valores que haga que las expresiones sean iguales, la consulta se considera resuelta y se devuelve el resultado, True.
* ***Inferencia***: corresponde a la aplicación de reglas lógicas para derivar nuevas proposiciones a partir de proposiciones previas.

Si comparamos los paradigmas que ya conocemos, Funcional v/s Lógico, tenemos que el paradigma funcional se basa en la evaluación de expresiones matemáticas y la teoría de funciones puras y se debe lograr la inmutabilidad evitando así los cambios de estados, estos últimos, solo se consiguen generando nuevas estructuras de datos en lugar de modificar las existentes. Por otro lado el paradigma de programación lógico que se basa en la lógica matemática y la teoría de conjuntos, el cual se compone de un conjunto de hechos y reglas lógicas los que se usan para responder consultas mediante la inferencia y lo que se conoce como proceso de unificación. A través de ello, se buscan todas las respuestas posibles mediante la lógica de inferencia utilizando una programación declarativa.

De lo anterior podemos decir que, la programación lógica se basa en la lógica y la inferencia, mientras que la programación funcional se basa en la evaluación de expresiones matemáticas y la composición de funciones.

# Descripción del problema

Un sistema de archivos es una parte fundamental de un sistema operativo, corresponde a un conjunto de procesos, métodos y reglas que utiliza un sistema operativo para administrar el almacenamiento de datos en la memoria de una computadora. Este sistema permite la organización, seguridad y localización de archivos, además de otras operaciones.

A lo largo de este informe se desarrollaran los requerimientos funcionales que permitan simular un sistema de archivos, donde un usuario pueda realizar acciones tales como agregar o eliminar contenido, así como también crear, modificar y eliminar archivos o carpetas, según permisos asignados, generando rutas de acceso a los archivos y carpetas, cambiarse de ruta, crear usuarios y drivers, cambiarse de unidad, iniciar y cerrar sesión.

# Análisis del problema

Descripción breve del paradigma y los conceptos del mismo que se ven aplicados en este proyecto (en sus palabras - max 1/2 página) (5%)

Análisis del problema respecto de los requisitos específicos que deben cubrir (max 1 página) (10%)

el análisis del problema: consiste en analizar los distintos requerimientos, no solo listarlos.

debes analizar todos los requerimientos, no "etc"

# Diseño de la solución

Diseño de la solución (presentar su enfoque de solución, describir, diagramar, descomposición de problemas, algoritmos o técnicas empleados para problemas particulares, recursos empleados) (max. 1 página. Puede incluir diagramas como anexos fuera del límite de las 5 páginas) (30%)

debes explicar las decisiones tomadas en cuanto al diseño, que TDAs, que representación, que capas, que algoritmo diseñaste

el diseño de la solución: es como se podría/como se implementó en el laboratorio

El diseño de la solución está basado en el lenguaje de programación Prolog, que debe simular los comandos básicos de operación de un sistema de archivos a través de consola de comandos.

# 

# Consideraciones de implementación

Aspectos de implementación (estructura del proyecto, bibliotecas empleadas, razones de su elección, compilador o interprete usado, etc.) (max 1/5 página) (10%)

no hay. Acá tienes que exponer la estructura del proyecto, versión de racket utilizada,

# Para la implementación de este laboratorio se recomienda la utilización del entorno virtual de prolog <https://swish.swi-prolog.org/> para realizar las pruebas de funcionamiento del código entregado.

# Instrucciones de uso

Para la ejecución de los algoritmos propuestos, debe abrir el archivo blablá, seleccionar todo el contenido, copiar y pegar en el editor virtual de prolog, <https://swish.swi-prolog.org/>

Para hacer las pruebas de funcionamiento debe ingresar en el apartado para las consultas del intérprete de prolog, lo siguiente:

* Ejemplos
* Resultados esperados
* Errores

Complementar con [ejemplos](https://uvirtual.usach.cl/moodle/mod/folder/view.php?id=156612) detallados en anexos que están fuera del límite de 5 páginas) (15%)

que funciones ocupaste?

Para cargar los módulos de la solución creada primero debes abrir el entorno de <https://swish.swi-prolog.org/> y cargar los módulos necesarios:

# Resultados y evaluación

Resultados y autoevaluación: Resumen de resultados obtenidos. Esto puede abordarse listando todos los requerimientos del proyecto en una tabla indicando el grado de alcance. Indicar que tipos de pruebas se hicieron. Cuantas de las pruebas fueron exitosas, cuantas fracasaron, razones de fallos. Especificar que funciones no se completaron y por qué no se completaron. (máximo 1/5 página) (10%)

no es copiar una foto de la ejecución, se pide evaluar los requerimientos implementados, los casos de prueba, para cuando funciona, para cuando no, cuantas pruebas realizaste, cuantas fueron exitosas, cuantas no

Tras evaluar los algoritmos de cada requerimiento funcional desarrollado, aplicando los script entregados por la asignatura y los script propios que se crearon para el desarrollo del laboratorio, podemos decir que la línea de programación elegida para la implementación, está acorde al modelo declarativo – lógico solicitado y los resultados obtenidos fueron los esperados como se aprecia en la tabla 1.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| RF Nº | Requerimiento funcional | Grado de alcance | Tipo o cantidad de pruebas | Cantidad de éxitos | Cantidad de fracasos | Razones de fallo |
| 1 | TDAs |  |  |  |  |  |
| 2 | system - constructor | 1 |  |  |  |  |
| 3 | system - addDrive | 1 |  |  |  |  |
| 4 | system - register | 1 |  |  |  |  |
| 5 | system - login | 1 |  |  |  |  |
| 6 | system - logout | 1 |  |  |  |  |
| 7 | system - switch-drive | 1 |  |  |  |  |
| 8 | system - mkdir | 0,5 |  |  |  |  |
| 9 | system- cd |  |  |  |  |  |
| 10 | system - add-file |  |  |  |  |  |
| 11 | system - del |  |  |  |  |  |
| 12 | system - copy |  |  |  |  |  |
| 13 | system - move |  |  |  |  |  |
| 14 | system - ren |  |  |  |  |  |

**Tabla 1.-** Muestra de

# Conclusiones

Conclusiones del trabajo (respecto de los alcances, limitaciones, dificultades de usar el paradigma para abordar el problema. Para los informes 2 en adelante, contrastar resultados alcanzados con el paradigma de turno con los anteriores) (máximo 1/2 página) (10%)

falta concluir sobre el trabajo realizado, que problematicas tuviste bajo el contexto de este problema, filesystem que problemas tuviste, como lo solucionaste, como se diferencia esta paradigma de lo que actualmente conoces, cómo se diferencia con otros lenguajes que conoces, técnicas, etc. Hay que ser técnicos

Para concluir este informe puedo decir que el objetivo del laboratorio 1 de la asignatura de paradigmas de la programación, correspondiente a la aplicación del paradigma declarativo – funcional esta parcialmente resuelto, ya que la implementación se desarrolló en base a lo solicitado y los resultados obtenidos fueron los esperados.

Tras el análisis, podemos deducir que el tiempo dedicado a realizar este laboratorio no fue el suficiente para desarrollar, implementar y evaluar todos los requerimientos funcionales solicitados, por lo que su funcionamiento a cabalidad no pudo ser testeado.

# Referencias

Referencias usando estilo de referencias APA (2.5%)

**Páginas web con contenido estático**

Apellido, A., Apellido, B., y Apellido, C. (20 de mayo de 2020). Título del artículo de la página web. Nombre del sitio web. <https://url.com>

**Páginas web con actualizaciones frecuentes**

Apellido, A., Apellido, B., y Apellido, C. (20 de mayo de 2020). Título del artículo de la página web. Nombre del sitio web. Recuperado el día mes año de https://url.com

no hay, mínimo debe estar el enunciado del laboratorio y/o las clases, de donde aprendiste el paradigma o de donde se obtuvo la información relativa al laboratorio?

Moodle del curso de paradigmas:

https://uvirtual.usach.cl/moodle/course/view.php?id=10036&section=17

Presentaciones del curso de paradigmas:

Ayuda de prolog:

<https://www.swi-prolog.org/pldoc/doc_for?object=atomic_list_concat/3>

Manual de prolog

<https://swish.swi-prolog.org/p/Tutorial%20de%20prolog.swinb>

Uso de módulos en prolog

https://www.swi-prolog.org/pldoc/man?section=modules