Portada xd

Indice

[**Descripción del proyecto** 3](#_Toc72241226)

[Propósito y alcance del proyecto 3](#_Toc72241227)

[Análisis de requerimientos y descripción de los principales *Test Cases* 3](#_Toc72241228)

[Descripción del PROCESO general seguido para el desarrollo del proyecto 3](#_Toc72241229)

[**Descripción del lenguaje** 3](#_Toc72241230)

[Nombre del lenguaje 3](#_Toc72241231)

[Descripción genérica de las principales características del lenguaje 3](#_Toc72241232)

[Listado de los errores que pueden ocurrir 3](#_Toc72241233)

[En compilación 3](#_Toc72241234)

[En ejecución 3](#_Toc72241235)

[**Descripción del compilador** 3](#_Toc72241236)

[Equipo de cómputo, lenguaje y utilerías especiales usadas 3](#_Toc72241237)

[Descripción del análisis léxico 3](#_Toc72241238)

[Patrones de construcción (expresados con expresiones regulares) de los elementos principales 3](#_Toc72241239)

[Enumeración de los “tokens” del lenguaje y su código asociado (ej: {: “left bracket”) 3](#_Toc72241240)

[Descripción del análisis de sintaxis 4](#_Toc72241241)

[Gramática formal empleada para representar las estructuras sintácticas (sin “codificar”) 4](#_Toc72241242)

[Descripción de generación de código intermedio y análisis semántico 4](#_Toc72241243)

[Código de operación y direcciones virtuales asociadas a los elementos del código 4](#_Toc72241244)

[Diagramas de sintaxis con las acciones correspondientes marcadas sobre ellos (puntos neurálgicos) 4](#_Toc72241245)

[Breve descripción de cada una de las acciones semánticas y de generación de código (no más de 2 líneas) 4](#_Toc72241246)

[Tabla de consideraciones semánticas (combinaciones factibles y errores de tipo) 4](#_Toc72241247)

[Descripción detallada del proceso de Administración de memoria en la compilación 4](#_Toc72241248)

[Especificación gráfica de CADA estructura de datos usada (Dir. Func. , tablas de vars, cuadruplos, pilas) 4](#_Toc72241249)

[**Descripción de la máquina virtual** 4](#_Toc72241250)

[Equipo de cómputo, lenguaje y utilerías especiales usadas ( en caso de ser diferente que el compilador) 4](#_Toc72241251)

[Descripción detallada del proceso de Administración de memoria en ejecución (arquitectura) 4](#_Toc72241252)

[Especificación gráfica de CADA estructura de datos usada para el manejo de scopes (Memoria local, global, etc.…) 4](#_Toc72241253)

[Asociación hecha entre las direcciones virtuales (compilación) y reales (ejecución) 4](#_Toc72241254)

[**Pruebas del funcionamiento del lenguaje** 5](#_Toc72241255)

[Incluir pruebas que “comprueben” el funcionamiento del proyecto 5](#_Toc72241256)

[Codificación de la prueba (en AGRO) 5](#_Toc72241257)

[Resultados arrojados por la generación de código intermedio y por la ejecución. 5](#_Toc72241258)

[**Documentación del código del proyecto** 5](#_Toc72241259)

# **Descripción del proyecto**

## Propósito y alcance del proyecto

Este proyecto es desarrollado con el propósito de acreditar la materia TC3048 (Diseño de compiladores) y conocer más a fondo el comportamiento verdadero de los lenguajes de programación que usamos en el día a día de nuestras vidas como desarrolladores de software.

El producto es un lenguaje orientado a objetos, con las características básicas de un lenguaje multiparadigma con conceptos también procedurales, al estilo de C++.

## Análisis de requerimientos y descripción de los principales *Test Cases*

## Descripción del PROCESO general seguido para el desarrollo del proyecto

# **Descripción del lenguaje**

## Nombre del lenguaje

El lenguaje que desarrollamos tiene el nombre de AGRO.

## Descripción genérica de las principales características del lenguaje

## Listado de los errores que pueden ocurrir

### En compilación

### En ejecución

# **Descripción del compilador**

## Equipo de cómputo, lenguaje y utilerías especiales usadas

Para el proyecto, utilizamos el generador de compiladores COCO/R para C#

## Descripción del análisis léxico

### Patrones de construcción (expresados con expresiones regulares) de los elementos principales

### Enumeración de los “tokens” del lenguaje y su código asociado (ej: {: “left bracket”)

## Descripción del análisis de sintaxis

### Gramática formal empleada para representar las estructuras sintácticas (sin “codificar”)

## Descripción de generación de código intermedio y análisis semántico

### Código de operación y direcciones virtuales asociadas a los elementos del código

### Diagramas de sintaxis con las acciones correspondientes marcadas sobre ellos (puntos neurálgicos)

### Breve descripción de cada una de las acciones semánticas y de generación de código (no más de 2 líneas)

### Tabla de consideraciones semánticas (combinaciones factibles y errores de tipo)

## Descripción detallada del proceso de Administración de memoria en la compilación

### Especificación gráfica de CADA estructura de datos usada (Dir. Func. , tablas de vars, cuadruplos, pilas)

# **Descripción de la máquina virtual**

## Equipo de cómputo, lenguaje y utilerías especiales usadas ( en caso de ser diferente que el compilador)

## Descripción detallada del proceso de Administración de memoria en ejecución (arquitectura)

### Especificación gráfica de CADA estructura de datos usada para el manejo de scopes (Memoria local, global, etc.…)

### Asociación hecha entre las direcciones virtuales (compilación) y reales (ejecución)

# **Pruebas del funcionamiento del lenguaje**

## Incluir pruebas que “comprueben” el funcionamiento del proyecto

### Codificación de la prueba (en AGRO)

### Resultados arrojados por la generación de código intermedio y por la ejecución.

# **Documentación del código del proyecto**