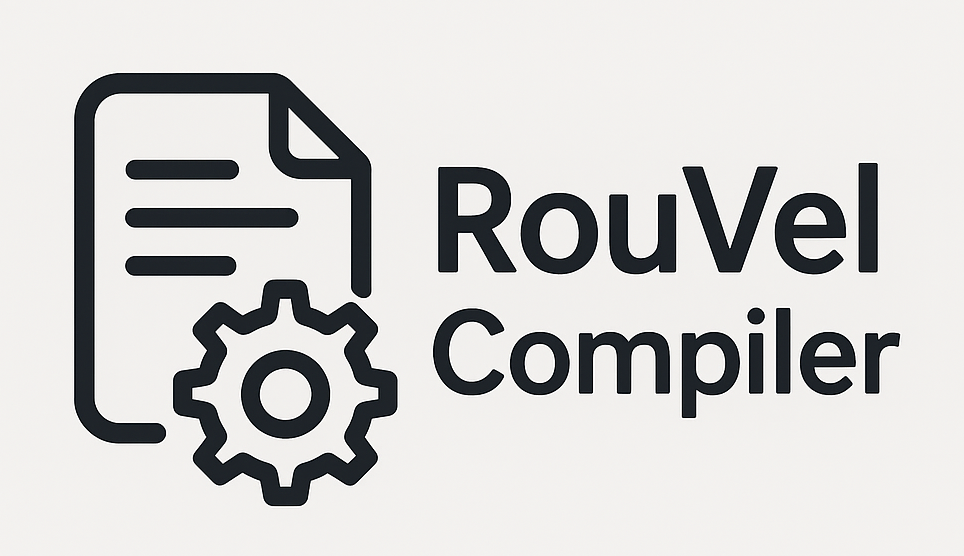
# RouVel Compiler

Trabajo práctico de cursada para la materia Compiladores e Intérpretes.

A partir de un lenguaje simple proporcionado por la cátedra, se implementó su compilador.



Ícono

## Índice

* [Requisitos](resources/markdown.md)
* [Características del lenguaje](resources/markdown/language.md)
* [Analizador léxico](resources/markdown/lexer.md)
* [Analizador sintáctico](resources/markdown/parser.md)

## Ejecución

Los binarios ya incluyen todo lo necesario para ejecutar el compilador. Al ejecutar el binario por primera vez se construirá la imagen Docker si no existe.

El programa funciona tanto con rutas absolutas como relativas. En resources/testFiles hay códigos de ejemplo.

### Linux

./rouvel-linux resources/testFiles/example0.uki

[!WARNING] De no haberle otorgado permisos de administrador a Docker, o no haber ejecutado el *script* con sudo, el ejecutable podría dar un error. Se recomienda realizar la primera acción, explicada en la sección de requisitos.

### macOS

./rouvel-macos resources/testFiles/example0.uki

### Windows

rouvel.exe resources/testFiles/example0.uki

## Eliminación de la Imagen (opcional)

docker rmi rouvel-compiler

Con similar se refiere a que use parsing ascendente y demás. Diferentes entradas para 3 y -3.

Sí constantes de tipo float, pero NO VARIABLES.

// 2>&1 es para redirigir la salida de error (warnings y errores) a la salida estándar. ./rouvel-linux resources/testFiles/example0.uki > text.txt 2>&1