# INSTITUTO TECNOLÓGICO DE COSTA RICA

SEDE CARTAGO

ESCUELA:

INGENIERIA EN COMPUTACIÓN

CURSO:

COMPILADORES E INTERPRETES IC – 5701

PROYECTO I PARTE II

PROFESOR:

ESTEBAN ARIAS MENDEZ

INTEGRANTES:

ANNIA GOMEZ BLANCO 201108349

FERNANDO UGARTE UGALDE 201048339

SEGUNDO SEMESTRE 2016

NOVIEMBRE DEL 2016



**Abstract**—Rust is a systems programming language that runs blazingly fast, prevents segfaults, and guarantees thread safety.

Many modern languages provide this memory safety, but they do it at runtime with references and garbage collection. JavaScript, Java, Ruby, Python, and Perl all fall into this camp. Rust is different. Rust is a statically typed compiled language meant to target the same tasks that you might use C or C++ for today, but it’s whole purpose in life is to promote memory safety. By design, Rust code can’t have dangling pointers, buffer overﬂows, or a whole host of other memory errors. Any code which would cause this literally can’t be compiled. The language doesn’t allow it. The Rust Programming Language. Also known as “The Book”, The Rust Programming Language is the most comprehensive resource for all topics related to Rust, and is the primary official document of the language.

**Contenido**

[Portada 1](#_Toc468220299)

[**Abstract** 2](#_Toc468220300)

[Introducción 4](#_Toc468220301)

[Objetivos 5](#_Toc468220302)

[Justificación del lenguaje de programación 6](#_Toc468220303)

[Herramientas Empleadas 7](#_Toc468220304)

[Marco teórico 8](#_Toc468220305)

[Desarrollo 9](#_Toc468220306)

[Análisis de Resultados 10](#_Toc468220307)

[Conclusiones 11](#_Toc468220308)

[Apéndice 1 12](#_Toc468220309)

[Apéndice 2 13](#_Toc468220310)

[Referencias 14](#_Toc468220311)

## **Introducción**

Recordamos que un compilador es un programa informático que traduce un programa escrito en un lenguaje de programación a otro lenguaje de programación, generando un programa equivalente que la máquina será capaz de interpretar, además que un intérprete es un programa informático capaz de analizar y ejecutar otros programas. Los intérpretes se diferencian de los compiladores o de los ensambladores en que mientras estos traducen un programa desde su descripción en un lenguaje de programación al código de máquina del sistema, los intérpretes sólo realizan la traducción a medida que sea necesaria, típicamente, instrucción por instrucción, y normalmente no guardan el resultado de dicha traducción.

En este caso se realizara el intérprete para el lenguaje de programación Rust este lenguaje surgió de un proyecto personal desarrollado por Graydon Hoere (trabajador de Mozilla). Empezó a trabajar en el en el año 2006, Mozilla se involucró en el proyecto en el año 2009 y lo dio a conocer en el año 2010. En ese mismo año paso del compilador inicial escrito en Ocaml al compilador auto-contenido, escrito en sí mismo en Rust. Se compilo así mismo 2011. La primera versión apareció en enero del 2012. La forma más fácil de probar Rust es a través del corralito, una aplicación en línea para escribir y ejecutar código Rust. Si desea probar Rust en su sistema, instálelo y vaya a través del tutorial del libro Guessing Game.

Así que para los programadores de sistemas Rust parece una gran opción sin embargo hay que recordar que es un lenguaje muy similar a C por lo tanto es de bajo nivel, probablemente diferente a lo que estamos acostumbrados hoy día. En otras Rust ofrece el rendimiento normalmente visto solo en los idiomas de sistemas de bajo nivel pero la mayoría de las veces, ¡sin duda se siente como un lenguaje de alto nivel! Para el público la primera versión apareció en enero del 2012.

## **Objetivos**

* Comparar el lenguaje Rust con sus principales sucesores y otros lenguajes pares a Rust.
* Definir ampliamente las características más fuertes del lenguaje Rust.
* Clasificar el lenguaje Rust dentro de un espectro amplio de lenguajes de programación para detallar mejor entender mejor su comportamiento y buen funcionamiento.
* Explicar cómo hace Rust su trabajo internamente.
* Describir ampliamente los usos e implementos que se le da a Rust en la actualidad.

## **Justificación del lenguaje de programación**

La más conveniente para el grupo de trabajo fue crear un intérprete para el lenguaje de programación que previamente fue asignado en clase en este caso específico fue asignado el lenguaje de programación llamado Rust.

Esta investigación y programación se realizó con el propósito de cumplir con los objetivos del curso de compiladores e intérpretes de este modo afianzar lo aprendido en clase y aprender a utilizar herramientas como Lex&Yacc.

Esta investigación se realizó aplicando la teoría aprendida en la clase antes mencionada, realizando un intérprete para el lenguaje de programación Rust para ello se tradujo al español la gramática de dicho lenguaje utilizando para desarrollar el programa el IDE Netbeans además de otras herramientas como GitHub. Todo desarrollado en el sistema operativo Ubuntu.

## **Herramientas Empleadas**

Las herramientas empleadas se detallan a continuación:

* Desarrollo principal de la aplicación fue realizado en NetBeans 8.2 para Ubuntu, con el uso de las extensiones Jflex 1.6.1
* LaTex desde su versión web https://www.sharelatex.com a través del navegador Google Chrome.
* Microsoft Office 2016 (word)
* GitHub desktop y la versión web https://github.com/frug54/Compilador

## **Marco teórico**

Algunas definiciones importantes para el análisis completo del documento serian:

* **Lenguaje:** Según la definición teórica, como lenguaje se entiende a un sistema de comunicación que posee una determinada estructura, contenido y uso. La programación es, en el vocabulario propio de la informática, el procedimiento de escritura del código fuente de un software. De esta manera, puede decirse que la programación le indica al programa informático qué acción tiene que llevar a cabo y cuál es el modo de concretarla.
* **Gramática:** La gramática es un ente formal para especificar, de una manera finita, el conjunto de cadenas de símbolos que constituyen un lenguaje". La gramática genera o describe un lenguaje.
* **BNF o EBNF:** La notación de Backus-Naur, (BNF) es un metalenguaje usado para expresar gramáticas libres de contexto: es decir, una manera formal de describir lenguajes formales. El BNF se utiliza extensamente como notación para las gramáticas de los lenguajes de programación de la computadora, de los sistemas de comando y de los protocolos de comunicación. Y la Extended Backus–Naur Form (EBNF), Que hacen más expresivas las gramáticas cualquier cosa que haga en EBNF se puede hacer en BNF
* **Scanner:** Es la primera fase de un compilador consistente en un programa que recibe como entrada el código fuente de otro programa (secuencia de caracteres) y produce una salida compuesta de tokens o símbolos. Estos tokens sirven para una posterior etapa del proceso de traducción, siendo la entrada para el analizador sintáctico.
* **Parser:** Es la parte de la compilación que transforma su entrada en un árbol de derivación.
* **Analizador semántico:** Genera la estructura base sobre la que podemos hacer funcionar otros módulos del sistema.
* **Compilador:** Programa informático que traduce un programa escrito en un lenguaje de programación a otro lenguaje de programación, generando un programa equivalente que la máquina será capaz de interpretar
* **Interprete:** Un intérprete es un software que recibe un programa en lenguaje de alto nivel, lo analiza y lo ejecuta. Para analizar el programa completo, va traduciendo sentencias de código y ejecutándolas si están bien, así hasta completar el programa origen. Los intérpretes informáticos, al contrario que los compiladores, no generan un fichero ejecutable u otro programa equivalente en otro lenguaje, por lo que cada vez que se ejecuta el programa original debe pasar por la fase de análisis.

## **Desarrollo**

analizador sintácticoasdadasasdasdasd asdasda sdas dasd asdasd

## **Análisis de Resultados**

Sintácticoasdadasasdasdassadasd d

## **Conclusiones**

Sintácticoasdadasasdasdassadasd

## **Apéndice 1**

Sintácticoasdadasasdasdassadasd

## **Apéndice 2**

Sintácticoasdadasasdasdassadasd

## **Referencias**

[1] Autores: Julián Pérez Porto y María Merino. Publicado: 2009. Actualizado: 2012.Definicion.de: Definición de lenguaje de programación (http://definicion.de/lenguaje-de-programacion/)