



Instituto Tecnológico de Costa Rica

Campus Tecnológico Central Cartago

Compiladores e Intérpretes

Parser

Erika Marin Schumann

Kenneth Palacios Molina - 2020035407

Jose Pablo Quesada Rodríguez - 2020211670

Kendall Fabian Guzman Ramirez- 2019076561

II Semestre

2023

Indice

Introducción:	3
Estrategia de solución:	4
Análisis de Resultados:	5
Corrección del Analizador Léxico	5
generación de la sintaxis	6
Lecciones aprendidas:	6
Casos de prueba:	7
Bitacora:	7
Manual de Usuario:	8
Bibliografía:	8

Introducción:

En esta sección del proyecto, se buscará generar un analizador sintáctico(parser), apoyándonos tanto de la librería jcup como jflex, librerías que permiten generar analizadores sintácticos y léxicos, respectivamente.

La utilización del software permitirá cargar un archivo de texto(.txt) y permitirá la lectura e interpretación de los contenidos de esta, el analizador léxico, fue la primera parte de este proyecto, el cual se encarga de leer el documento y clasificar los caracteres por tokens específicos.

Ahora el analizador sintáctico (**parser**) creado en este proyecto como segundo avance, tiene la capacidad de detectar errores sintácticos en el documento proporcionado, e indicarlos en una ventana de interfaz de usuario.

Para la implementación de esto se deberá definir en el código del mismo, tokens requeridos, creación de un léxico y una interfaz de usuario comprensible y fácil de utilizar(**alta usabilidad**).

Estrategia de solución:

Para darle solución a este se necesitaba construir un analizador lexico y sintactico, de acuerdo a los objetivos solicitados en el proyecto de mostrar un listado de los errores encontrados tanto léxicos como sintácticos en una interfaz de usuario,

Se empezó el proyecto corrigiendo los errores que se tuvieron en la entrega pasada del analizador léxico, ya que era necesario para este proyecto también, entonces se completaron las partes faltantes del anterior proyecto, siendo estas partes:

- Identificación de Flotantes
- Identificación de caracteres con #
- Corrección de números científicos

Para el analizador sintáctico, la parte de este proyecto se implementó la librería jcup, la cual permite generar automáticamente archivos necesarios para el funcionamiento de esta, la ejecución de este archivo solicita la existencia de una gramática que se tiene que definir, como un archivo para ser consumido por la librería y que generen los archivos .java requeridos

Todos los resultados obtenidos, deben ser manipulados por una interfaz, por esto en esta se definió en la creación del frame, la forma en la cual en esta se cargaran los datos txt

Análisis de Resultados:

Corrección del Analizador Léxico

Objetivos	No logrado	Logrado
		100%
Identificación de Flotantes		100%
Identificación de caracteres con #		100%
Corrección de números científicos		100%
Diseño de interfaz		100%
detección de los diferentes símbolos desde el parser por medio de un switch		100%

generación de la sintaxis

Objetivos	No logrado	Logrado
		100%
Generación de estructura principal de un programa		100%
Generación de no terminales para la sección de constantes		100%
Generación de no terminales para la sección de Variables Globales		100%
Generación de no terminales para la sección de Funciones		100%
Generación de no terminales para la parte del Main		70%
Generación de no terminales para las estructuras de control		70%

Lecciones aprendidas:

- Entendimiento más profundizado de herramientas como lo son jcup y jflex para la creación de lo solicitado, analizador lexico y sintactico
- Se comprendió mejor el funcionamiento interno de un compilador, sobre todo la parte del analizador sintáctico
- El uso de gramáticas (no terminales y terminales)
- Se mejoró el trabajo en equipo, mediante la practica de actividades en conjunto y la división de actividades

Casos de prueba:

Bitacora:

Fecha	Objetivos Realizados
28/10/2023	<ul style="list-style-type: none">• Se arreglaron las expresiones regulares para detectar comentarios multilínea• Se agregó la expresión regular para detectar cuando se abre un comentario multilínea pero no se cierra.
30/10/2023	<ul style="list-style-type: none">• Se terminaron de crear los documentos lexerCup.flex
1/11/2023	Nos enteramos que strings ("texto") no se daban válidos si se encontraban en varias líneas, entonces se creó la expresión regular para retornar error en dicho caso

2/11/2023	Se hicieron las expresiones para los numeros en notacion cientifica con distintos errores como nada después de l
4/11/2023	La detección de errores se logró hacer funcionar y los números de notación científica, por lo que se arregló la detección de caracteres (#13) por ejemplo y de tokens con largo > 128
5/11/2023	Se empezó a implementar el apartado de interfaz de usuario, Dividido en las secciones requeridas, para mostrar separado el analizador lexico y sintactico, acorde a los objetivos solicitados
7/11/2023	Se completó la interfaz de usuario al 100%

Manual de Usuario:

Bibliografía:

Charlead. (2019). JCup y JFlex | Analizador sintáctico con Java (explicación paso a paso)

[YouTube Video]. Retrieved from <https://www.youtube.com/watch?v=4Z6Tnit810Y>

