**Instituto Tecnológico de Costa Rica**

**Departamento de Computación**

**Sede Alajuela**

**Curso:**

Compiladores e Intérpretes

**Profesor:**

Aurelio Sanabria

**Integrantes:**

Carlos Adán Arguello Calderón - 201173805

Jefri Cárdenas Villatoro - 2013101392

**Tipo de evaluación:**

Laboratorio 1

## Introducción

Este laboratorio consiste en la realización de un “scanner” para la realización de un compilador completo, para esto es importante entender la definición de tokens, para qué sirven las expresiones regulares y manejar bloques de tipo de definición (contiene un conjunto palabras). En este primer avance se busca comprender la primer etapa del proceso de compilación y este se encarga de obtener los tokens presentes en el texto de entrada.

**Desarrollo del lenguaje y su funcionamiento**

Para el desarrollo de este “scanner” se necesitó aplicar conocimientos de Python y programación orientada a objetos. Se implementaron los siguientes métodos:

* **init:** Es donde se inicializa la clase agrupando los tokens por grupos según su contexto.
* **agregar:** Lee un archivo de texto, y envía todos los bloques encontrados y no encontrados a una lista vacía.
* **generador\_tokens:** Hace un recorrido de la lista para verificar de qué tipo es el token y también si hay errores, que son si no contiene un bloque de la gramática y si quedan caracteres sin poder tokenizar, estos se mencionan en la ejecución.
* **imprimir\_tokens:** Este método se encarga de imprimir los tokens una vez se haya finalizado el recorrido del programa.

**Gramática EBNF inicial y gramática transformada**

La gramática EBNF inicial elaborada se realizó pensando de una vez en su transformación por lo que se encuentra en su forma transformada. Esta gramática está diseñada a partir de ejemplos de direcciones que se pueden dar a las personas para llegar desde la ubicación actual a otro punto. Esta se encuentra en una versión comprimida y sencilla en la que se tiene que expresar la dirección de la manera definida por la gramática para su correcta aplicación la cual sería la siguiente:

**Inicio ::= [Del | De la]**

**Lugar ::= “[a-z | ]+”**

**Distancia ::= [1-9][0-9]+**

**ValorMedidas::= [cuadras | metros | kilometros]**

**Conectores ::= [y | hasta | si no | hacia el | hacia la | hasta la]**

**Cardinales::= [norte | sur | este | oeste | arriba | abajo | derecha | izquierda]**

**Detalles ::= [a-z | ]+**

**Direccion ::= Inicio Lugar Detalles Distancia ValorMedidas Conectores**

**Cardinales .**

**DireccionTica ::= Direccion+**