INSTITUTO TECNOLÓGICO DE MÉRIDA

INGENIERÍA EN SISTEMAS COMPUTACIONALES

LENGUAJES Y AUTÓMATAS 1

INVESTIGACIÓN FINAL U5 Y U6

SEMESTRE: SEXTO

GRUPO: 6SB

ALUMNO:

Joshua Jesus Aviles Interian

FECHA DE ENTREGA:

24/05/2024

U5. Análisis Sintáctico

Un analizador sintáctico es un programa diseñado para identificar si una o varias secuencias de caracteres pertenecen a un lenguaje específico, usualmente lenguajes libres de contexto. Los analizadores se dividen en dos tipos según cómo construyen los nodos del árbol de derivación sintáctico: ascendentes y descendentes.

Definición de Gramáticas Libres del Contexto (GLC)

Las gramáticas libres del contexto, también conocidas como gramáticas de tipo 2, generan lenguajes que pueden ser reconocidos por autómatas de pila, ya sean determinísticos o no. Estas gramáticas se definen mediante una cuádrupla G = (N, T, P, S):

- N: Conjunto finito de símbolos no terminales.
- T: Conjunto finito de símbolos terminales, donde N y T no tienen elementos en común.
- P: Conjunto finito de producciones, representadas como relaciones en N × (N ∪ T)*.
- S: Símbolo inicial de la gramática.

Las producciones en una GLC tienen la forma A → w, donde A es un símbolo no terminal y w es una cadena de símbolos terminales y no terminales, pudiendo incluir la cadena vacía (ε). A partir de un símbolo no terminal, se aplican reglas de producción para derivar una cadena compuesta únicamente por símbolos terminales, conocida como frase o lexema. Las cadenas formadas por símbolos terminales y no terminales se llaman formas de frase.

Representación de una Expresión Regular en una GLC

Para representar una expresión regular como a*b+ en una GLC, definimos:

N: {E, A, B} (E es el símbolo inicial).

T: {a, b} (símbolos terminales).

P: Producciones definidas como:

E → A | B

A → aA | ε (representa a*)

B → bB | b+ (representa b+)

Para derivar la cadena "aaabbb", seguimos estos pasos:

E → A (usando la producción 1)

A → aA (usando la producción 2 tres veces)

A → ε (usando la producción 2 una vez)

B → bB (usando la producción 1)

B → bB (usando la producción 3 tres veces)

B → b (usando la producción 3 una vez)

El resultado es la cadena "aaabbb".

Árbol de Derivación

Un árbol de derivación es una representación gráfica de un proceso de derivación en una gramática, con las siguientes características:

La raíz del árbol es el símbolo inicial de la gramática.

Los nodos interiores están etiquetados con símbolos no terminales.

Las hojas están etiquetadas con símbolos terminales.

Para cada nodo interior etiquetado con A, sus hijos corresponden a una producción A → X1X2...Xn de la gramática, donde Xi puede ser un símbolo terminal o no terminal.

Ejemplo de gramática: G = (Σ = {a, b}, N = {S, A, B}, P, S) con P: S → aABAa, A → ε | aA, B → ε | bB. La construcción del árbol de derivación para la palabra "aa" sigue estas reglas.

Propiedades de un árbol de derivación incluyen que la raíz es un símbolo no terminal, cada hoja corresponde a un símbolo terminal o a λ, y cada nodo interior corresponde a un símbolo no terminal.

U6. Máquinas de Turing

Definición

Una Máquina de Turing es un modelo matemático que define formalmente lo que significa un algoritmo o función computable. Introducida por Alan Turing en 1936, es fundamental en la teoría de la computación y sirve para explorar los límites de lo que puede ser calculado.

Una Máquina de Turing consta de:

- Cinta infinita dividida en celdas.
- Cabeza de lectura/escritura que se mueve a lo largo de la cinta.
- Conjunto de reglas que determinan su comportamiento en cada paso.

Características

- Modelo de cálculo: Puede simular cualquier algoritmo computacional.
- Cinta infinita: Actúa como memoria para leer y escribir símbolos.
- Conjunto finito de estados: Incluye un estado inicial y estados de aceptación.
- Cabezal de lectura/escritura: Se mueve sobre la cinta para leer y escribir.
- Función de transición: Dicta cómo cambia el estado y qué acción se realiza (escribir, moverse).

Estructura

Una Máquina de Turing incluye:

Cinta: Infinita en ambas direcciones, con celdas que contienen símbolos de un alfabeto finito Γ, incluyendo el símbolo especial de blanco (#).

Cabezal de lectura/escritura: Se mueve una celda a la vez, leyendo y escribiendo símbolos.

Conjunto de estados (Q): Incluye un estado inicial (q0) y estados de aceptación (F).

Alfabeto de entrada (Σ): Contiene los símbolos de entrada ($\Sigma \subset \Gamma$ y no incluye el símbolo de blanco).

Función de transición (δ): Define las operaciones de la máquina (cambio de estado, escritura de símbolo, movimiento del cabezal).

Ejemplo

Consideremos una Máquina de Turing que decide si una cadena binaria contiene un número par de unos:

Estados: {q0, q1, qaccept, qreject}

Alfabeto de entrada: {0, 1}

Alfabeto de la cinta: {0, 1, #}

Estado inicial: q0

Estados de aceptación: {qaccept}

Función de transición:

δ (q0, 0) = (q0, 0, R)

δ (q0, 1) = (q1, 1, R)

δ (q1, 0) = (q1, 0, R)

δ (q1, 1) = (q0, 1, R)

δ (q0, #) = (qaccept, #, R)

δ (q1, #) = (qreject, #, R)

Para la entrada "1100":

Inicial: q0, cadena: "1100#", cabeza en el primer símbolo.

Paso 1: δ (q0, 1) = (q1, 1, R), nueva cadena: "1100#", cabeza en el segundo símbolo.

Paso 2: δ (q1, 1) = (q0, 1, R), nueva cadena: "1100#", cabeza en el tercer símbolo.

Paso 3: δ (q0, 0) = (q0, 0, R), nueva cadena: "1100#", cabeza en el cuarto símbolo.

Paso 4: δ (q0, 0) = (q0, 0, R), nueva cadena: "1100#", cabeza en el símbolo de blanco.

Paso 5: δ (q0, #) = (qaccept, #, R), la máquina acepta la cadena porque contiene un número par de unos.

Referencias

[1] Linz, P. (2016). Introducción a los Lenguajes Formales y Autómatas (6ta ed.). Jones & Bartlett Learning.

[2] Lewis, H. R., & Papadimitriou, C. H. (1998). Elementos de la Teoría de la Computación (2da ed.). Prentice Hall.

[3] Kozen, D. C. (1997). *Autómatas y Computabilidad*. Springer.

[4] Martin, J. C. (2010). *Introducción a los Lenguajes y la Teoría de la Computación* (4ta ed.). McGraw-Hill.