

Nombre: Ana Lucía Salvatierra
Martínez.

ID: 2386

Grado: 5to bachillerato.

Materia: computación

Teoría de autómatas

Un autómata finito determinista (AFD) es un modelo matemático que se utiliza para reconocer lenguajes regulares.

Usos de los AFD

- Reconocimiento de lenguajes regulares
- Análisis léxico en compiladores
- Validación de entradas de usuario
- Modelado de sistemas con estados finitos
- Optimización de código

Ejemplos

- Reconocimiento de patrones en cadenas de texto
- Análisis de protocolos de comunicación
- Diseño de sistemas de control

Gramáticas regulares

Son gramáticas que generan lenguajes regulares, los cuales pueden ser reconocidos por autómatas finitos.

Ejemplo

Se utiliza para definir patrones de búsqueda (expresiones regulares) en lenguajes como Python o javascript

Gramáticas recursivas

Incluyen producciones que permiten la recursividad, generando lenguajes más complejos

Ejemplo

Lenguajes libres de contextos

Diferencias clave: las gramáticas regulares son limitadas a estructura lineales mientras que las cursivas manejan jerarquías y anidamiento.

Análisis sintáctico

Análisis sintáctico descendente (top-down):

Comienza con el símbolo inicial de la gramática y construye el árbol de análisis expandiendo producciones hasta llegar a los tokens del programa

Análisis sintáctico ascendente (bottom-up):

Parte de los tokens del programa y reduce las producciones hasta llegar al símbolo inicial

Arboles de análisis sintáctico

- Definición : representa jerarquía que muestra como se estructuran las sentencias del programa según la gramática del lenguaje
- Utilidad: verifica el código cumpla con la sintaxis del lenguaje.

Análisis léxico

Tokenización: es el proceso de dividir un texto de entrada (código fuente) llamadas tokens. En la compilación, el analizador léxico (lexer) convierte la secuencia de caracteres en una secuencia de tokens.

Expresiones regulares: se utilizan para definir patrones que describen como deben lucir los tokens. En el análisis léxico, las expresiones regulares y clasificación

Reconocimiento de la palabra clave: las palabras clave (keywords) son edificadores reservados en un lenguaje de programación (como if,while ,int).

Identificación de tokens: los tokens de clasifican en categorías como
identificadores (variables, funciones)
Palabras clave(reservadas del lenguaje)
Operadores (aritméticos,lógicos)

Algoritmos de análisis de cadenas

- **Algoritmo de boyer-moore:** es un eficiente algoritmo de búsqueda de cadenas que localiza la ocurrencia de un patrón (cadena objetivo) dentro de un texto

Aplicación:

- Búsqueda de cadenas en textos largos
- Optimización de programas que requieren localizar substratos de manera eficiente
- Editores de texto motores de búsqueda para mejorar el rendimiento

Compiladores en intérpretes

Estructura de un compilador: es un programa que traduce código fuente escrito en un lenguaje de alto nivel a código máquina o lenguaje ensamblador. Sus componentes incluyen el analizador léxico, sintático, semántico, generador de código y optimización.

Fases del compilador

- **Análisis léxico:** divide el código fuente en tokens
- **Análisis sintáctico:** verifica la estructura gramatical del código (árbol sintático)
- **Generador de código:** traduce el código a un intermedio a código máquina
- **Optimización de código:** mejora el código generado para hacerlo más eficiente

Lenguajes utilizados en la creación de compilación Lex herramienta para generar analizadores léxicos **yacc**

Lenguaje de programación c

Stdio.h: librería estándar de entrada y salida en c.

Cómic.h: librería no estandar (común en entornos dos/Windows) para control de entrada/salida de consola

Lista de librerías comunes en c

1. Stdio.h: entrada/salida estándar
2. Comió.h: funciones de consola (no estandar)
3. Stdlib.h: funciones generales
4. Sting.h: manipulación de cadenas
5. Math.h: funciones matemáticas
6. Tome.h: manejo de fechas y tiempos
7. Ctype.h: clasificación de caracteres
8. Assert.h macros de depuración
9. Eterno.h manejo de errores
10. Signal.h: manejo de señales

Manejadores de formatos para tipos de datos en c %d entero decimal (int), %f flotante (float), %if doble precisión.

Aplicaciones prácticas de automatas

Reconocimiento de patrones :

En diferentes áreas, como procesamiento de imágenes, análisis de señales o bioinformática.

Procesamiento de lenguaje natural:

Y su uso en inteligencia artificial, permitiendo analizar y generar textos automáticos.

Análisis de texto y minería de datos:

Destacando su importancia en el análisis de información para extraer conocimientos útiles.