

- 1.- Alfabetos, hileras, con-catenación. Lenguajes, operaciones con lenguajes, propiedades asociativas y distributivas de la concatenación.
- 2.- Gramáticas, jerarquía de Chomsky. Gramáticas no restringidas, de contextp-sensitiva, monotónicas, forma normal de Kuroda, de contexto libre, normal de Chomsky, de Greibach, lineales, de izquierda y derecha. Teoremas sobre gramáticas.
- 3.- Teoría de autómatas. Autómata finito, de pila, ligado linealmente, máquina de Turing. Relaciones entre lenguajes generados por una grá-mática y aceptados por un autómata. Teoremas sobre autómatas.
- 4.- Problema del "Parsing". Algoritmos de reconocimiento. Gramáticas de contexto libre y algoritmo de Earley. Otros tipos de gramáticas usua-les en compiladores: de precedentes, LL(k), LR(k), de precedencia de operador, de contexto limitado, de precedencia de estrategia mixta simple.
- 5.- Compiladores, compaginadores, intérpretes, Estructura de un compila-dor. Compilador de un paso, de varios pasos. El explorador, construc-ción del explorador. Sistemas de construcción de exploradores.
- 6.- El analizador sintáctico, función y programación. Tablas de símbolos y técnicas de acceso, búsqueda binaria, direccionamiento Hash, es-structuras encadenadas, arboles, bloques, red. Datos en la tabla de símbolos, descriptores.
- 7.- La asignación de memoria, la organización de la memoria, asignación dinámica, su representación, las plantillas. Diversas técnicas.
- 8.- El generador de código, la rutinas semánticas. Formas internas, ope-radores, notación polaca, cuartetos, tercetos, arboles, bloques. Trans-formación de la notación.
- 9.- Técnicas de optimización de códigos. Optimización de operaciones arit-méticas, bloques y ciclos. Metacompiladores, algunos ejemplos sencil-los Prepro cesadores. Optimización de compiladores, algunos casos simples.