

INDICE

PREFACIO	XIII
AGRADECIMIENTOS	XVII
ADVERTENCIA PARA EL ESTUDIANTE	XIX
1. INTRODUCCION	1
1.1 Estructura de la máquina	2
1.2 Evolución de los componentes de un sistema de programación	4
1.2.1 Ensambladores	4
1.2.2 Cargadores	4
1.2.3 Macros	6
1.2.4 Compiladores	6
1.2.5 Sistemas formales	7
1.3 Evolución de los sistemas operativos	7
1.4 El sistema operativo desde el punto de vista del usuario: funciones	9
1.5 El sistema operativo desde el punto de vista del usuario: lenguaje de control de lotes	11
1.6 El sistema operativo desde el punto de vista del usuario: recursos	13
2. ESTRUCTURA DE LA MAQUINA, LENGUAJE DE MAQUINA Y LENGUAJE ENSAMBLADO	19
2.1 Estructura general de las máquinas	19
2.1.1 Cómo familiarizarse con una máquina nueva	21
1. Memoria.- 2. Registros.- 3. Datos.-	
4. Instrucciones.- 5. Características especiales.-	
2.1.2 Estructura de las máquinas 360 y 370	23
2.2 Lenguaje de máquina	32
2.2.1 Por el camino largo, sin ciclos	33
2.2.2 Modificación de las direcciones con el uso de instrucciones como datos	35
2.2.3 Modificación de las direcciones con el uso de registros índices	37
2.2.4 Formación de ciclos	38
2.3 Lenguaje ensamblador	39
2.3.1 Un programa en lenguaje ensamblador	40
2.3.2 Ejemplo con literales	42

3. ENSAMBLADORES	54
3.1 Procedimiento general de diseño	55
3.2 Diseño del ensamblador	55
3.2.1 Planteo del problema	55
3.2.2 Estructura de los datos	57
3.2.3 Formato de las bases de datos	60
3.2.4 Algoritmo	67
3.2.5 Organización modular	71
3.3 Procesamiento de tablas: búsqueda y clasificaciones	74
3.3.1 Búsqueda lineal	74
3.3.2 Búsqueda binaria	75
3.3.3 Clasificación	78
3.3.4 Búsqueda al azar	85
4. MACROLENGUAJES Y MACROPROCESADORES	103
4.1 Macroinstrucciones	104
4.2 Particularidades de los macros	106
4.2.1 Argumentos de macroinstrucción	106
4.2.2 Expansión condicional de macros	109
4.2.3 Macrollamadas dentro de macroinstrucciones	111
4.2.4 Macroinstrucciones que definen macros	113
4.3 Implementación	114
4.3.1 Implementación restringida: algoritmo de dos pasos	115
4.3.2 Un algoritmo de una pasada	121
4.3.3 Implementación de macrollamadas dentro de macros	124
4.3.4 Implementación dentro de un ensamblador	127
5. CARGADORES	141
5.1 Esquemas de carga	142
5.1.1 Cargadores "compile y transfiera"	142
5.1.2 Esquema general de carga	142
5.1.3 Cargadores absolutos	143
5.1.4 Encadenamiento de subrutinas	146
5.1.5 Cargadores reasignadores	148
5.1.6 Cargadores de encadenamiento directo	151
5.1.7 Otros sistemas de carga - Encuadernadores, cargadores encadenadores, "overlays", encuadernadores dinámicos	155
5.2 Diseño de un cargador absoluto	158
5.3 Diseño de un cargador de encadenamiento directo	160
5.3.1 Planteo del problema	160
5.3.2 Especificación de las estructuras de datos	167
5.3.3 Formato de las bases de datos	169
5.3.4 Algoritmo	172
6. LENGUAJES DE PROGRAMACION	188
6.1 Importancia de los lenguajes de alto nivel	188
6.2 Peculiaridades de un lenguaje de alto nivel	190
6.3 Tipos de datos y estructuras de datos	191
6.3.1 Series de caracteres	191
6.3.2 Series de bits - Operadores booleanos	193
6.3.3 Estructuras de datos	193
6.4 Asignación de almacenamiento y alcance de nombres	194
6.4.1 Clases de almacenamiento	194
6.4.2 Estructura de bloque	195

6.5 Flexibilidad de acceso	196
6.5.1 Indicador	196
6.5.2 Variables de rótulo y colecciones de rótulos	197
6.6 Modularidad funcional	199
6.6.1 Procedimientos	200
6.6.2 Recursión	200
6.7 Operación asíncrona	200
6.7.1 Condiciones	201
6.7.2 Señales	202
6.7.3 Multitarea	203
6.8 Extensibilidad y macros de tiempo de compilación	203
7. INTRODUCCION A LOS SISTEMAS FORMALES Y LOS LENGUAJES DE PROGRAMACION	214
7.1 Usos de los sistemas formales en los lenguajes de programación	214
7.1.1 Especificación de lenguajes	215
7.1.2 Compiladores sintácticos	215
7.1.3 Estudios de complejidad de estructuras	215
7.1.4 Análisis de estructuras	216
7.2 Especificación formal	216
7.2.1 Acceso a un formalismo	216
7.2.2 Desarrollo de una especificación formal	218
7.3 Gramáticas formales	220
7.3.1 Ejemplos de gramáticas formales	221
7.3.2 La derivación de sentencias	222
7.3.3 Formas sentenciales y sentencias	222
7.4 Jerarquía de lenguajes	223
7.5 Forma de Backus-Naur – Forma normal de Backus-BNF	225
7.6 Sistemas canónicos	226
7.6.1 Ejemplo: especificación de sintaxis	229
7.6.2 Especificación de una traducción	231
7.6.3 Algoritmo de reconocimiento y traducción	232
7.7 Sistemas canónicos y sistemas formales	235
8. COMPILADORES	248
8.1 Planteo del problema	248
8.1.1 Problema Nº 1 – Reconocimiento de los elementos básicos	249
8.1.2 Problema Nº 2 – Reconocimiento de las unidades sintácticas e interpretación del significado	250
8.1.3 Formas intermedias	251
8.1.4 Problema Nº 3 – Asignación de almacenamiento	254
8.1.5 Problema Nº 4 – Generación del código	256
8.1.6 Modelo generalizado del compilador	260
8.2 Fases del compilador	263
8.2.1 Fase léxica	263
8.2.2 Fase sintáctica	266
8.2.3 Fase de interpretación	270
8.2.4 Optimización	275
8.2.5 Asignación de almacenamiento	283
8.2.6 Generación de código	288
8.2.7 Fase de ensamble	295
8.2.8 Pasadas de un compilador	295
8.2.9 Sobre la parte 3	297

8.3 Estructuras de datos	297
8.3.1 Planteo del problema	298
8.3.2 Implementación	301
8.4 Enunciados de recursión, llamada y retorno	302
8.5 Clases de almacenamiento y su uso	306
8.5.1 Almacenamiento estático	307
8.5.2 Almacenamiento automático	308
8.5.3 Almacenamiento controlado interno	308
8.5.4 Almacenamiento controlado externo	309
8.5.5 Almacenamiento basado	309
8.6 Ejecución	309
8.6.1 Almacenamiento estático	309
8.6.2 Almacenamiento estático	310
8.6.3 Almacenamiento controlado y almacenamiento basado	311
8.7 Estructuras de bloque	312
8.7.1 Acceso a la información con estructuras de bloque	314
8.7.2 Asignación de almacenamiento para estructuras de bloque	315
8.8 Enunciados Go To no locales	319
8.9 Interrupciones	319
8.10 Indicadores (pointers)	320
9. SISTEMAS OPERATIVOS	329
9.1 Programación E/S: procesadores múltiples y mecanismos de interrupción ..	330
9.1.1 Evolución de los sistemas de procesador múltiple	330
9.1.2 Programación E/S	332
9.1.3 Estructura del procesador de E/S	333
9.1.4 Ejemplos de programas de E/S	335
9.1.5 Comunicaciones entre la UCP y el canal	337
9.1.6 Estructura de las interrupciones y procesamiento	338
9.1.7 Ejemplo de procesamiento de interrupciones de E/S (figura 9.9)	341
9.1.8 Procesadores múltiples (multiprocesadores)	345
9.2 Administración de la memoria	345
9.2.1 Asignación contigua simple	346
9.2.2 Asignación particional	346
9.2.3 Asignación particional reasignable	349
9.2.4 Asignación paginada	352
9.2.5 Asignación paginada por demanda	355
9.2.6 Asignación segmental	357
9.2.7 Asignación segmentada-paginada	364
9.3 Administración de los procesadores	364
9.3.1 Planificador	367
9.3.2 Control de tráfico	368
9.3.3 Condición de carrera	370
9.3.4 Empates	371
9.3.5 Sistemas de procesadores múltiples	372
9.4 Administración de dispositivos	378
9.4.1 Características de los dispositivos periféricos	378
9.4.2 Técnicas de administración de dispositivos	382
9.5 Administración de la información	383
9.5.1 Desarrollo de los sistemas de archivo	384
9.5.2 Estructura de un sistema de archivo general	385
9.5.3 Ejemplo de sistema de archivo	387
9.5.4 Particularidades de un sistema de archivo general	390
9.5.5 Revisión del modelo de sistema de archivo general	397

9.5.6 Segmentación	399
9.5.7 El sistema de archivo MULTICS y el modelo general	405
10. BIBLIOGRAFIA	416
 APENDICE A: Sistema 360 – Datos de referencia	425
 APENDICE B: Convenciones de encadenamiento	438
1. Reserva y restauración de los contenidos de registro	438
2. Area de reserva	438
3. Encadenamiento de sistema estándar	439
4. Ejemplo de encadenamiento	440
5. Implementación del procesamiento de argumentos	442