

## INDICE

|   |           |
|---|-----------|
| <b>Capítulo 1. El Turbo C++ y el Lenguaje C</b>                                   | <b>1</b>  |
| Introducción  | 1         |
| Introducción al turbo C++   | 2         |
| Instalación de la versión 3 del turbo C++   | 4         |
| Carga del editor turbo C++ para crear programas                                   | 6         |
| Opciones: File, Edit, Search, Run, Compile, Debug, Project; Options, Window, Help | 7         |
| Edición y ejecución de un programa ejemplo  | 8         |
| Compilación de un programa turbo C++  | 10        |
| La función MAIN ( )   | 13        |
| Funciones secundarias   | 15        |
| Punto y coma, llaves y bloques, explicaciones o comentarios                       | 16        |
| Resumen de instrucciones para crear un programa turbo C++                         | 20        |
| Resumen del capítulo y ejemplos varios  | 22        |
| Generalidades sobre el editor de turbo C++  | 24        |
| <b>Capítulo 2. Declaración e inicialización de variables</b>                      | <b>27</b> |
| Tipos de datos utilizados en un programa  | 27        |
| Declaración e inicialización de variables caracteres tipo char                    | 28        |
| Variables tipo INT  | 32        |
| Variables tipo FLOAT y DOUBLE   | 32        |
| Declaración de una variable   | 34        |
| Lugar donde se declaran las variables   | 38        |
| Identificador de variables  | 40        |
| Palabras reservadas en C y C ++   | 41        |
| Tipo de variables   | 42        |
| Variables globales y locales en turbo C ++  | 44        |
| <b>Capítulo 3. Funciones de entrada y salida</b>                                  | <b>47</b> |
| Printf ( )  | 47        |
| Secuencias de Escape  | 50        |
| Especificador para enteros sin signo  | 52        |
| Conversiones de números enteros en diferentes bases                               | 53        |
| Conversión de caracteres en su correspondientes código ASCII                      | 54        |
| Especificadores especiales para la sentencia PRINTF()                             | 57        |
| Longitud del campo impreso  | 59        |
| Modificadores para indicar el signo de un valor numérico                          | 61        |
| SCANF()   | 63        |
| Sintaxis de la función SCANF()  | 65        |
| Funciones de entrada/salida aplicarlas a un solo carácter: GET-CHAR(), PUTCHAR()  | 69        |
| Lectura de un fichero de texto  | 72        |
| Creación, lectura y copia de ficheros de texto                                    | 73        |
| Creación de un nuevo programa   | 74        |
| Visionado en pantalla de un programa ya existente                                 | 74        |
| Copia de un fichero ya existente  | 75        |
| Diferencias existentes entre Turbo C ++ y C                                       | 78        |
| Ejemplo sencillo de programación turbo C++  | 80        |
| Variables globales y locales en Turbo C++   | 81        |

|   |     |
|---|-----|
| <b>Capítulo 4. Sentencias de control del programa</b>             | 85  |
| El bucle WHILE  | 85  |
| Sintaxis de WHILE   | 90  |
| El bucle FOR()  | 91  |
| Sintaxis de la sentencia FOR()                                    | 95  |
| Bucles FOR() para Mostar arrays                                   | 99  |
| La sentencia IF()   | 102 |
| La función IF() aplicada a mas de dos opciones                    | 103 |
| Sintaxis de la sentencia IF()                                     | 104 |
| El SE-IF  | 106 |
| La sentencia SWICH()  | 109 |
| La sentencia BREAK  | 111 |
| La sentencia # DEFINE   | 112 |
| Sintaxis de la sentencia # DEFINE                                 | 114 |
| Sentencias para controlar los comandos # DEFINE y # INCLUDE       | 115 |
| Operadores relacionales o símbolos comparativos                   | 117 |
| Operadores lógicos  | 119 |
| El operador condicional (?:)                                      | 120 |
| Operador incremento(+ +) y decremento (- -)                       | 121 |
| Programas obsoletos escritos en Turbo C versión 2                 | 122 |
| <b>Capítulo 5. Punteros</b>                                       | 127 |
| Asignación de punteros que apuntan a números enteros              | 131 |
| Puntero entero flotante   | 134 |
| Asignación de punteros  | 137 |
| Puntero tipo flotante   | 140 |
| Ejemplo sobre matrices y punteros                                 | 149 |
| Polimorfismo: sobrecadena dentro de una cadena                    | 153 |
| Función que devuelve un puntero                                   | 155 |
| Plantilla de una estructura                                       | 157 |
| Asignación de valores y punteros a una estructura                 | 160 |
| Creación de una clase y de sus correspondientes objetos           | 162 |
| Entrada y salida de datos en turbo C ++                           | 167 |
| El mandato cin. Get ( )   | 169 |
| Un ejemplo clásico sobre clases y objetos                         | 171 |
| Variables públicas y privadas                                     | 176 |
| La herencia dentro de una clase                                   | 176 |
| El concepto enum para numerar una lista de datos                  | 180 |
| Creación de objetos en clases que contienen funciones matemáticas | 182 |
| Creación de varios objetos  | 184 |
| Creación de una clase amiga (Friend)                              | 185 |
| Herencia y clase amiga (Friend)                                   | 188 |
| Otro caso de función amiga (Friend)                               | 190 |
| Herencia múltiple y función amiga                                 | 192 |
| Transferencia de una estructura a una función                     | 195 |
| Transferencia de una estructura tipo array a una función          | 197 |
| Estructuras anidadas en una clase                                 | 200 |
| <b>Capítulo 6: Cadenas o Tiras de Caracteres</b>                  | 205 |
| Cadenas de caracteres (STRING)                                    | 205 |

|  |     |
|--|-----|
| Las tiras de caracteres en la memoria del ordenador                            | 208 |
| Cadenas estáticas o globales   | 214 |
| Asignación de tiras a un puntero   | 217 |
| Los punteros en los arrays de tiras  | 219 |
| Los punteros en los arrays de dos dimensiones                                  | 223 |
| Introducción de tiras en una matriz  | 230 |
| Funciones de entrada y salida de cadenas creadas por el propio usuario         | 231 |
| Escribe una tira de caracteres en pantalla                                     | 234 |
| Arrays de cadenas  | 236 |
| Manipulación de tiras  | 239 |
| Funciones que devuelven cadenas  | 242 |
| Matrices de punteros   | 245 |
| Ordenes incluidas en los elementos de una matriz                               | 246 |
| Matrices que contienen punteros  | 247 |
| Funciones estándar para manejar cadenas o Springs                              | 249 |
| La función Strcpy ()   | 250 |
| La función strcat ()   | 250 |
| La función Strscmp ()  | 251 |
| Ejercicios sobre funciones que manejan cadenas                                 | 252 |
| Terminador nulo y cadena invertida   | 253 |
| Punterio que apunta a otro puntero   | 255 |
| Funciones que definen el tipo de carácter teclado                              | 257 |
| <b>Capítulo 7. Funciones creadas para Facilitar el Trabajo del Programador</b> |     |
| Programa tipo con biblioteca C ++ incorporada                                  | 263 |
| LPRINT(): Función creada para imprimir instrucciones                           | 266 |
| Buffer para la empresa   | 271 |
| Movimiento del cursor en pantalla  | 275 |
| Introducción de datos por pantalla: SAYGET()                                   | 275 |
| Cadenas y subcadenas   | 282 |
| Derecha y centro de una cadena   | 286 |
| Selección de partes de una cadena  | 289 |
| Eliminar espacios vacíos situados en los extremos de la cadena                 | 296 |
| Ajuste de cadenas a los márgenes de su propio campo                            | 300 |
| Cadena formada por caracteres repetidos  | 303 |
| Código numérico (ASCII) de las teclas  | 305 |
| Color en pantalla  | 309 |
| <b>Capítulo VIII. Ficheros de datos en Turbo C ++</b>                          | 313 |
| Introducción   | 313 |
| Diferentes tipos de ficheros de datos TURBO C++                                | 315 |
| Turbosistema para manipulación de archivos en TURBO C++                        | 316 |
| Organización de los datos  | 318 |
| Asignación de nombres a los ficheros   | 319 |
| Creación de un fichero   | 320 |
| Apertura de un fichero de texto  | 321 |
| Funciones mas usuales para un fichero de disco                                 | 322 |
| Crear un fichero nuevo   | 327 |
| Nomenclatura general de ficheros creados                                       | 329 |

|  |     |
|--|-----|
| Características de los ficheros de base de datos creados por el usuario            | 329 |
| Sentencias utilizadas para crear un fichero  | 335 |
| Activar un fichero existente   | 336 |
| Descripción de la función USA(ARCHI)   | 337 |
| Introducción de los primeros datos en un fichero                                   | 340 |
| Introducción de campos de datos en un fichero                                      | 343 |
| Mostrar todos los registros de un fichero  | 347 |
| Hallar el numero de registros de un fichero  | 350 |
| Desglosar los registros en sus correspondientes campos                             | 354 |
| Agrupar los campos formando registros  | 357 |
| Tipos de campos  | 359 |
| Filtrado de errores en el campo fecha  | 363 |
| Invertir el formato fecha  | 369 |
| Copiar ficheros  | 369 |
| Copiar dos ficheros con distinto nombre  | 370 |
| Los argumentos de la función MAIN()  | 373 |
| Búsqueda de registros  | 375 |
| Corrección de registros  | 381 |
| Entrada y salida de ficheros   | 386 |
| Ordenación de registros  | 387 |
| Listado condicionado de ficheros   | 395 |
| Inserción de registros   | 398 |
| Borrar, depurar, blanquear ficheros  | 403 |
| Depurar los ficheros   | 407 |
| Blanquear ficheros   | 409 |
| El puntero en los ficheros   | 411 |
| Reemplazar campos de un registro   | 413 |
| Condicionales en la búsqueda de registros  | 417 |
| Seleccionar registros creando un nuevo fichero                                     | 424 |
| Modificar la estructura de un fichero  | 428 |
| Ejemplos de programación directa aplicando funciones creadas por el propio usuario | 437 |
| Resultados obtenidos   | 441 |
| <b>Capitulo IX. Ejemplo de programación estructurada por medio de menús</b>        | 445 |
| Introducción al programa de ejemplo: FASE 4.CPP                                    | 445 |
| Carga del programa FASE 4.CPP que se adjunta en este libro                         | 447 |
| Colores en pantalla  | 450 |
| Activación de FASE 4. CPP estructurado en forma de menú                            | 453 |
| Opción (R) nombre del fichero  | 455 |
| Opción (X) creación de un fichero  | 456 |
| Opción (J) inicia la entrada de datos  | 457 |
| Ampliar registros, ordenar y mostrar el fichero                                    | 458 |
| Opción (I) insertar registros ordenados  | 460 |
| Búsqueda de registros en modo interactivo  | 461 |
| Filtrado de registros  | 464 |
| Listado y configuración del programa FASE 4. CPP                                   | 466 |
| <b>Apéndice A. manual de instrucciones para la programación en TURBO C</b>         | 473 |

|  |     |
|--|-----|
| sentencias y funciones creadas   | 473 |
| Nomenclatura general   | 474 |
| Añadir registros   | 475 |
| Listado de registros   | 475 |
| Búsqueda de registros  | 476 |
| Hallar el numero de registros de un fichero  | 477 |
| Listar determinados campos de un fichero   | 477 |
| Reemplazo de los campos de un registro   | 477 |
| Búsqueda condicionada de registros   | 477 |
| Copiar a un nuevo fichero los registros seleccionados  | 477 |
| Desdoblar registros y agrupar campos   | 478 |
| <b>Apéndice B. Resumen General de Mandatos y Ejercicios en Turbo C ++</b>                    | 479 |
| Configuración de una estructura  | 479 |
| Declaración de una Strutt  | 480 |
| Acceso a los primeros miembros de una estructura   | 482 |
| Estructuras y punteros   | 483 |
| Creación de una clase y de sus correspondientes objetos operador de resolución (::)          | 485 |
| Clase definida con Strutt, unión o class   | 488 |
| Declaración de una Strutt  | 489 |
| Declaración de una unión   | 489 |
| Declaración de una class   | 489 |
| Entrada y salida de datos en Turbo C++   | 490 |
| Turbo C ++ y el mandato cin.get ()   | 492 |
| Sobre clases y objetos   | 492 |
| La herencia dentro de una clase  | 492 |
| El concepto enum para enumerar una lista de datos  | 494 |
| Funciones turbo C ++ que manejan cadenas   | 495 |
| La función strcpy ()   | 496 |
| La función Strcat ()   | 497 |
| La función strcmp ()   | 499 |
| Ejercicios sobre funciones que manejan cadenas   | 500 |
| Ejemplos prácticos sobre manejo de string  | 508 |
| Funciones que definen el tipo de carácter teclado  | 509 |
| Funciones para convertir matemáticas   | 513 |
| Funciones fecha y tiempo   | 520 |
| Funciones de entrada y salida (I/O) (E/S)  | 524 |
| Apéndice C. Paralelismo Existente entre las Funciones Turbo C++ Creadas y dBASE IV o Clipper | 529 |
| Índice temático  | 539 |