

# Contenido

<b>Prólogo</b> .....	<b>XIII</b>
<b>Capítulo 1. Introducción</b> .....	<b>1</b>
¿A quién va dirigido este libro? .....	2
¿Qué material se necesita? .....	2
¿Qué objetivo se persigue? .....	2
Bibliografía disponible sobre C++ .....	3
El problema de las versiones .....	3
Algunas notas sobre siglas .....	4
Brevísima historia de C++ .....	5
¿Es C++ una mera extensión de C? .....	5
Objetos: una nueva forma de pensar .....	5
Programación Orientada-A-Objetos .....	7
¿Ha muerto la programación estructurada? .....	8
¿Por qué C++? .....	9
¿Es la OOP el vértice de la programación? .....	11
<b>Capítulo 2. OOP: Conceptos básicos</b> .....	<b>13</b>
OOP: conceptos teóricos básicos .....	14
Envío de mensajes a objetos .....	15
Abstracción .....	18
Encapsulación .....	19
Abstracción y encapsulación en C++ .....	20
Herencia .....	24
Polimorfismo .....	27

<b>Capítulo 3. Donde C y C++ difieren</b> .....	<b>33</b>
Prototipado de funciones y chequeo de tipos en C++ .....	34
Nuevas palabras clave .....	35
Saltos sobre declaraciones .....	35
Inicialización de arrays .....	38
Conversión por asignación de punteros a void .....	38
Tipo de enlace en variables constantes globales .....	40
Enlace de tipo seguro .....	40
Ámbito de struct's .....	43
Constantes literales de caracteres .....	44
Tipo de los enumeradores .....	44
Macro <code>__cplusplus</code> .....	45
Definiciones múltiples .....	45
Espacio único de nombres de identificadores y estructuras .....	46
<b>Capítulo 4. Donde C++ se destaca</b> .....	<b>49</b>
Delimitadores de comentario .....	50
Operador cualificador de ámbito ( <code>::</code> ) .....	52
Declaraciones mezcladas con expresiones .....	54
Valores por defecto en parámetros formales .....	56
Referencias y paso de parámetros por referencia .....	60
El especificador "inline" .....	62
Sobrecarga de funciones .....	63
Los operadores "new" y "delete" .....	68
Migración de ANSI C a C++: reglas básicas .....	75
<b>Capítulo 5. Donde "la clase" se evidencia</b> .....	<b>77</b>
Algunas notas sobre estilos de codificación .....	78
Primer acercamiento a las clases .....	81
Definición de clase y miembros .....	86
Datos miembros de una clase .....	86
Ocultación de los datos .....	89
Funciones miembros: viejas sierras con dientes nuevos .....	92
El puntero implícito "this" .....	93
Funciones miembros constantes .....	96
Interfaz e implementación .....	98

<b>Capítulo 6. Entrada y salida de datos</b> . . . . .	<b>99</b>
Introducción a la librería "iostream.h" . . . . .	99
Mensajes i/o dirigidos a objetos "streams" . . . . .	102
Funciones miembros de los streams . . . . .	105
Manipuladores . . . . .	106
Formato de los datos de entrada/salida . . . . .	108
Un sencillo ejemplo práctico . . . . .	110
Manejo de ficheros . . . . .	111
 <b>Capítulo 7. Cuando irrumpen los constructores</b> . . . . .	 <b>113</b>
Inicialización de objetos . . . . .	114
Constructores . . . . .	117
Destruyores . . . . .	125
Un sencillo control . . . . .	126
 <b>Capítulo 8. De la objetividad de lo racional</b> . . . . .	 <b>127</b>
Definamos un nuevo tipo: la clase "Racional" . . . . .	127
Sobrecarga de operadores . . . . .	133
Los peligros de las sobrecargas . . . . .	138
Funciones amigas de una clase . . . . .	143
Sobrecarga de los operadores de inserción y extracción . . . . .	152
Operadores de conversión . . . . .	153
Funciones miembros estáticas . . . . .	157
Un epílogo de colusión . . . . .	160
Interfaz de la clase "Racional" . . . . .	161
Implementación de la clase "Racional" . . . . .	163
Manejo elemental de objetos de tipo "Racional" . . . . .	168
 <b>Capítulo 9. De la herencia y derivados</b> . . . . .	 <b>171</b>
Un primer acercamiento a la herencia . . . . .	172
Herencia simple: un lobo con piel de melocotón . . . . .	173
Derivación pública: la facilidad de la herencia . . . . .	175
Constructores de clases en jerarquías de derivación . . . . .	179
Inicialización de clases base . . . . .	180
Derivaciones no-públicas: sólo para tus ojos . . . . .	184
Conversiones estándar bajo derivación . . . . .	190
C + + : un país de maravillas . . . . .	195

<b>Capítulo 10. De las funciones virtuales</b>	<b>199</b>
Reutilización y refinamiento del código heredado	199
Selectores de tipo	203
Funciones virtuales	206
Cuestiones de ámbito en funciones virtuales	210
Cualificación de acceso de las funciones virtuales	214
Destructores virtuales	215
Resolución estática de funciones virtuales	217
Funciones virtuales puras y clases bases abstractas	219
Constructores virtuales	221
<b>Capítulo 11. Conclusión</b>	<b>223</b>
<b>Anexo 1. De las librerías y otros útiles</b>	<b>225</b>
Librerías de clases	225
Marcos de aplicación	229
Gestión de persistencia	242
<b>Anexo 2. Bibliografía comentada</b>	<b>249</b>
Libros básicos sobre C++	250
Libros de estilo en C++	253
Libros de programación avanzada en C++	255
Libros sobre software orientado a objetos	259
Libros de análisis y diseño orientado-a-objetos (OOA & OOD)	262
<b>Anexo 3. OOA &amp; OOD, una aproximación crítica</b>	<b>271</b>
Software orientado-a-objetos	271
Cinco criterios sobre modularidad en OOSS	273
Seis principios esenciales en OOSS's	274
Bottom-Up versus Top-Down	276
Ejemplos de conceptualización objetual	277
Ciclo de vida de un OOSS	278
Transición del OOA al diseño orientado-a-objetos	278
Diseño orientado-a-objetos	279

Clasificación de los métodos de OOD . . . . .	281
Diagramanía . . . . .	282
Modelado de jerarquías . . . . .	283
Diseño orientado-a-objetos de bases de datos . . . . .	285
Algunas herramientas OO-Upper-CASE y OO-I-CASE . . . . .	286
Referencias . . . . .	287
<b>Anexo 4. Insistente persistencia . . . . .</b>	<b>289</b>
Fundamentos conceptuales . . . . .	290
Impulsión de persistencia . . . . .	292
Sistemas persistentes . . . . .	293
La persistencia según Borland . . . . .	295
La función escritora . . . . .	298
Función lectora . . . . .	305
Ejemplos prácticos . . . . .	313
Persistencia recursivo-derivativa (RDP) . . . . .	314
La evolución de la persistencia . . . . .	319
Referencias . . . . .	320