

INDICE

Prefacio	VII
Capitulo 1. Desarrollo software	1
1.1. Ingeniería del software	2
Profesionalismo y ética	
1.2. Calidad del software	3
Corrección	
Fiabilidad	
Robustez	4
Usabilidad	
Mantenibilidad	
Reusabilidad	5
Portabilidad	
Eficiencia	6
Problemas relativos a la calidad	
1.3. Modelos del ciclo de desarrollo	7
El modelo en cascada	9
El modelo en espiral	10
Modelo de desarrollo evolutivo	
1.4. El lenguaje UML	11
Relaciones UML	13
1.5. Tratamiento de errores	15
1.6. Análisis de algoritmos	16
Funciones de crecimiento y notación $O()$	17
Comparación de funciones de crecimiento	18
Análisis de la ejecución de bucles	19
Bucles anidados	20
1.7. Ingeniería del software y estructuras de datos	21
Capitulo 2. Diseño orientado a objetos	25
2.1. Panorámica de la orientación a objetos	
2.2. Utilización de objetos	26
Abstracción	27
Creación de objetos	28
2.3. Bibliotecas de clases y paquetes	29
La declaración import	30
2.4. Estado y comportamiento	31
2.5. Clases	32
Datos de la instancia	34
2.6. Encapsulación	
Modificadores de visibilidad	35
Datos locales	36
2.7. Constructores	
2.8. Sobrecarga de métodos	37
2.9. Mas cosas sobre las referencias	
La referencia nula	39
La referencia this	40
Alias	41
Depuración de memoria	43

Paso de objetos como parámetros	
2.10. El modificador STATIC	
Variables estáticas	44
Métodos estáticos	
2.11. Clases envoltorio	45
2.12. Interfaces	46
La interfaz comparable	
La interfaz iterator	47
2.13. Herencia	
Clases derivadas	48
El modificador protected	50
La referencia super	
Sustitución de métodos	51
2.14. Jerarquías de clases	
La clase object	52
Clases abstractas	54
Jerarquías de interfaces	55
2.15. Polimorfismo	
Referencias y jerarquías de clases	56
Polimorfismo mediante herencia	57
Polimorfismo mediante interfaces	59
2.16. Tipos genéricos	60
2.17. Excepciones	
Mensajes de excepción	61
La instrucción Try	62
Propagación de excepciones	
La jerarquía de clases de excepciones	63
Capítulo 3. Colecciones	71
3.1. Introducción a las colecciones	72
Tipos abstractos de datos	73
La API Collections de Java	74
3.2. Una colección de tipo conjunto	75
Interfaces	76
Iteradores	78
Excepciones	
3.3. Utilización de un conjunto: bingo	79
3.4. Implementación de un conjunto mediante matrices	82
Gestión de la capacidad	83
3.5. La clase ArraySet	84
Las operaciones size e isEmpty	
La operación add	87
La operación addAll	89
La operación removeRandom	90
La operación remove	91
La operación union	
La operación contain	92
La operación equals	93
La operación iterator	94
La operación toString	96

Descripción UML	
3.6. Análisis de ArraySet	
Análisis de add	
Análisis de remove	
Análisis de removeRandom	
Análisis de addAll	98
Análisis de union	
Análisis de equals	
Capítulo 4. Estructuras enlazadas	103
4.1. Las referencias como enlace	104
4.2. Gestión de listas enlazadas	
Inserción de nodos	106
Borrado de nodos	107
Nodos ficticios	
4.3. Elementos sin enlace	108
Listas doblemente enlazadas	
4.4. Implementación de un conjunto mediante enlaces	109
La clase LinkedSet	110
La operación add	112
La operación removeRandom	113
La operación remove	114
La operación iterator	115
4.5. Análisis de LinkedSet	
Análisis de add	
Análisis de removeRandom	118
Capítulo 5. Black Jack	123
5.1. Un juego de Black Jack	
5.2. Diseño inicial	124
5.3. Implementación de un juego de Black Jack	
La clase Card	129
La clase Deck	131
La clase Hand	135
La clase BlackJack	138
La clase BlackJackGUI	141
La clase BlackJackDemo	148
6.2. Utilización de pilas para la evaluación de expresiones postfijas	155
6.3. Utilización de pilas para recorrer un laberinto	161
6.4. Implementación de pilas con enlaces	
La operación push	166
La operación pop	168
Otras operaciones	
6.5. Implementación de pilas con matrices	169
La operación push	
La operación pop	170
Otras operaciones	
6.6. Implementación de pilas mediante la clase	171
Java.util.stack	
Operaciones distintivas	
Herencia e implementación	172

6.7. Análisis de las implementaciones de pilas	
Análisis de push	173
Análisis de pop	174
Capítulo 7. Colas	179
7.1. Tipo abstracto de datos para una cola	180
7.2. Utilización de las colas para cifrado	182
7.3. Utilización de las colas para simular un despacho de entradas	185
7.4. Utilización de una cola para una ordenación raíz	189
7.5. Implementación de colas mediante enlaces	193
La operación enqueue	194
La operación dequeue	195
Otras operaciones	196
7.6. Implementación de las colas mediante matrices	
La operación enqueue	197
La operación dequeue	198
Otras operaciones	199
7.7. Implementación de las colas mediante matrices circulares	200
7.8. Análisis de las implementación de las colas	203
enqueue	204
dequeue	205
Capítulo 8. Listas	211
8.1. Tipo abstracto de datos para una lista	212
8.2. Utilización de listas ordenadas para calendarios de torneos	219
8.3. Utilización de listas indexadas en el problema de Josefo	223
8.4. Implementación de listas mediante matrices	
La operación remove	226
La operación contains	
La operación add para una lista ordenada	229
Operaciones particulares de las listas desordenadas	
Operaciones particulares de las listas indexadas	230
8.5. Implementación de listas mediante enlaces	
La operación remove	231
Listas doblemente enlazadas	232
8.6. Análisis de las implementaciones de lista	235
Análisis de las implementaciones de lista ordenadas	236
Análisis de las implementaciones de lista desordenadas	237
Análisis de las implementaciones de lista indexada	239
Capítulo 9. Calculadora	245
9.1. Una calculadora	
9.2. Diseño inicial	246
9.3. Implementación de una calculadora	249
La clase infixToPostfix	251
La clase listPostfixEvaluator	255
La clase calculatorGUI	256
La clase calculatorDemo	261
Capítulo 10. Recursión	265
10.1. Pensamiento recursivo	266
La recursión en matemáticas	267
10.2. Programación recursiva	268

Recursión e iteración	269
Recursión directa e indirecta	270
10.3. Utilización de la recursión	
Recorrido de un laberinto	271
Las torres de Hanoi	275
10.4. Análisis de los algoritmos recursivos	279
Capítulo 11. Ordenación y búsqueda	287
11.1. Búsqueda	
Búsqueda lineal	288
Búsqueda binaria	289
Comparación entre los algoritmos de búsqueda	292
11.2. Ordenación	293
Ordenación por selección	295
Ordenación por inserción	297
Método de la burbuja	298
Ordenación rápida	300
Ordenación por mezcla	303
Capítulo 12. Árboles	309
12.1. Árboles	310
Clasificaciones de los árboles	311
12.2. Estrategias para la implementación de árboles	
Estrategias computacional para la implementación de árboles mediante matrices	312
Estrategias de enlaces simulados para la implementación de árboles basada en matrices	313
Análisis de los árboles	314
12.3. Recorrido de un árbol	
Recorrido descendente	315
Recorrido en árbol	
Recorrido ascendente	316
Recorrido por orden de niveles	317
12.4. Implementación de árboles binarios	318
El método removeLeftSubtree	323
El método find	324
El método iteratorInOrder	325
12.5. Utilización de árboles binarios para construir árboles de expresión	326
Capítulo 13. Árboles de búsqueda binaria	339
13.1. Un árbol de búsqueda binaria	340
13.2. Implementación de árboles de búsqueda binaria mediante enlaces	341
La operación addElement	342
La operación removeElement	344
La operación removeAllOccurrences	346
La operación removeMin	347
13.3. Utilización de árboles de búsqueda binaria: implementación de listas ordenadas	348
Análisis de la Implementación de BinarySearchTreeList	350
13.4. Árboles de búsqueda binaria equilibrados	351
Rotación a la derecha	352

Rotación a la izquierda	353
Rotación a la derecha-izquierda	
Rotación a la izquierda-derecha	
13.5. Implementación de árboles de búsqueda binaria mediante árboles AVL	354
Rotación a la derecha en un árbol AVL	
Rotación a la izquierda en un árbol	355
Rotación a la derecha en un árbol AVL	
Rotación izquierda derecha en un árbol AVL	356
13.6. Implementación de arbole de búsqueda binaria mediante árboles rojo/negro	356
Inserción en una árbol rojo/negro	358
Eliminación de un elemento en un árbol ro/negro	361
13.7. Implementación de árboles de búsqueda binaria mediante la API Collections de Java	362
Capitulo 134. Árbol genealógico	373
14.1. Árbol genealógico	
14.2. Diseño inicial	374
14.3. Implementación de un árbol genealógico	377
La clase Person	
La clase AncestorTreeNode	379
La clase AncestorTree	381
La clase AncestorTorGUI	388
La clase AncestorTorDemo	400
Capitulo 15. Cúmulos	403
15.1. Cúmulos	404
La operación addElement	405
La operación removeMin	
La operación findMin	407
15.2. Utilización de cúmulos para ordenadores por cúmulo	
15.3. Utilización de cúmulos para implementar colas de prioridad	408
15.4. Implementación de cúmulos mediante enlaces	411
La operación addElement	412
La operación removeMin	414
La operación findMin	
15.5. Implementación de cúmulos mediante matrices	416
La operación addElement	417
La operación removeMin	418
La operación findMin	419
15.6. Análisis de las implementaciones de cúmulos	
La operación addElement	
La operación removeMin	420
La operación findMin	
Ordenación por cúmulo	421
Capitulo 16. Árboles de búsqueda multivía	425
16.1. Combinación de los conceptos de árboles	
16.2. Árboles 2-3	426
Inserción de elementos en un árbol 2-3	

Eliminación de elementos en un árbol 2-3	428
16.3. Árboles 2-4	431
16.4. Árboles B	
16.5. Motivación para el desarrollo de los árboles B	433
Árboles B*	
Árboles B+	434
Análisis de los árboles B	435
16.6. Estrategias de implementación para los árboles	
Capítulo 17. Funciones hash	441
17.1. Almacenamiento hash	442
17.2. Funciones hash	
El método de división	444
El método de plegado	
El método del cuadrado medio	
El método de la transformación base	445
El método de análisis de dígitos	
El método dependiente de la longitud	446
Funciones hash en el lenguaje Java	
17.3. Resolución de colisiones	
Encadenamiento	447
Direccionamiento	449
17.4. Borrado de elementos en una tabla hash	
Borrado de un elemento en una implementación encadenada	452
Borrado de un elemento en una implementación de direccionamiento abierto	453
17.5. Tablas hash en la API Collections de Java	
La clase Hashtable	455
La clase HashSet	456
La clase HashMap	457
La clase Hashtable IdentityHashMap	458
La clase WeakHashMap	460
LinkedHashSet y LinkedHashMap	461
Capítulo 18 Grafos	467
18.1. Grafos no dirigidos	468
18.2. Grafos dirigidos	469
18.3. Redes	470
18.4. Algoritmos comunes para los grafos	471
recorridos	472
Pruebas de conectividad	475
Árbol de recubrimiento mínimo	476
Determinación de la ruta mas corta	477
18.5. Estrategias para la implementación de Grafos	
Listas de adyacencia	480
Matrices de adyacencia	481
Capítulo 19. Buscador web	487
19.1. Buscador web	
19.2. Diseño inicial	488
19.3. Implementación de u buscador web	489
La clase WebCrawler	491

La clase WebCrawlerGUI	497
La clase WebCrawlerDemo	499
Indice	503