

Índice general

1	Vectores y celdas enlazadas	1
1.1	Introducción	1
1.2	Vectores en memoria dinámica	1
1.2.1	Vectores de tamaño cero y punteros nulos	2
1.2.2	Ejemplos de reservar/liberar estructuras de vectores	3
1.2.3	Almacenamiento automático vs dinámico	9
1.2.4	Rangos en un vector	12
1.3	Celdas enlazadas	13
1.3.1	Recorrer una lista	14
1.3.2	Insertar y borrar una celda	15
1.3.3	Celda controlada desde la anterior	17
1.3.4	Rango en una lista de celdas	18
1.4	Vectores vs celdas enlazadas	19
1.5	Problemas	20
2	Análisis avanzado de eficiencia	23
2.1	Introducción	23
2.2	Análisis de algoritmos recursivos	24
2.2.1	Ecuación recurrente	24
2.2.2	Ecuación recurrente simplificada	25
2.3	Deducción de la expresión general	26
2.3.1	Cambio de variable	28
2.4	Ecuación característica	30
2.4.1	Recurrencias Homogéneas Lineales	31
2.4.2	Recurrencias no Homogéneas	33
2.4.3	Ecuaciones recurrentes no lineales: cambios de variable	35
2.5	Análisis amortizado	40
2.5.1	Calculando por fuerza bruta	41
2.5.2	Método contable	45

2.5.3	Funciones de potencial	48
2.5.4	Mezcla de operaciones	52
2.6	Problemas	57
3	Pilas y colas	59
3.1	Introducción	59
3.2	La clase <i>Pila</i>	60
3.2.1	Implementaciones	62
3.2.2	Implementación con vector de tamaño fijo	63
3.2.3	Implementación con vector de tamaño variable	64
3.2.4	Implementación basada en celdas enlazadas	66
3.3	La clase <i>Cola</i>	70
3.3.1	Implementaciones	72
3.3.2	Implementación basada en vectores	72
3.3.3	Implementación basada en celdas enlazadas	78
3.4	Modificaciones sobre tipos <i>Pila</i> y <i>Cola</i>	81
3.4.1	Plantillas	81
3.4.2	Mejoras en la interfaz	82
3.4.3	Mejoras en la implementación	83
3.4.4	Colas limitadas	86
3.5	La clase <i>Cola doble</i>	88
3.5.1	Implementaciones	89
3.5.2	Implementación basada en celdas enlazadas	90
3.5.3	Implementación con vectores por bloques	92
3.6	La clase <i>Cola con prioridad</i>	100
3.6.1	Implementación	103
3.6.2	Orden de prioridad como parámetro de plantilla	105
3.7	Problemas	106
4	Listas	109
4.1	Introducción	109
4.2	La clase <i>Lista</i>	110
4.2.1	Posiciones en una lista	110
4.2.2	Interfaz propuesta	111
4.3	Implementación del tipo <i>Lista</i>	117
4.3.1	Representaciones de los tipos	119
4.3.2	Implementación de iteradores	120
4.3.3	Inserción y borrado	121
4.4	La clase <i>ListaSimple</i>	122
4.5	Implementación de <i>ListaSimple</i>	123
4.5.1	Implementación con celdas simplemente enlazadas	123
4.5.2	Implementación con celdas simplemente enlazadas y cabecera	128

4.6	La clase <i>ListaFwd</i>	132
4.6.1	Inserción y borrado	133
4.7	Implementación de <i>ListaFwd</i>	135
4.7.1	Inserción y borrado	136
4.8	Detalles avanzados de implementación	136
4.8.1	Constructores para el tipo <i>Celda</i>	137
4.8.2	Construcción y asignación por movimiento	138
4.8.3	Representación de la cabecera	140
4.8.4	Gestión de excepciones	142
4.8.5	Constructor por movimiento y <i>noexcept</i>	143
4.9	Iteradores y programación genérica	145
4.9.1	Abstracción por iteración	145
4.9.2	Tipos de iteradores	146
4.10	Problemas	148
5	Árboles generales	151
5.1	Introducción y terminología básica	151
5.1.1	Definiciones básicas	152
5.1.2	Recorridos	155
5.2	Representación para árbol general	160
5.2.1	Representación con celdas enlazadas	161
5.2.2	Moviéndose por los nodos del árbol	162
5.2.3	Creación y modificación de árboles	163
5.2.4	Liberar un árbol completo	165
5.2.5	Listar elementos de un árbol general	167
5.3	La clase <i>ArbolGen</i>	170
5.3.1	Interfaz pública del tipo <i>ArbolGen</i>	171
5.3.2	Implementación del tipo <i>ArbolGen</i>	175
5.3.3	Iteradores sobre un árbol general	178
5.4	E/S de árboles. Serialización	182
5.4.1	Árbol desde recorridos	182
5.4.2	Preorden más profundidad	183
5.4.3	Preorden con marcas o centinelas	184
5.4.4	Preorden/postorden: principio y fin	185
5.5	Problemas	187
6	Árboles n-arios	189
6.1	Introducción y terminología básica	189
6.1.1	Definiciones básicas	190
6.2	Representación de árboles n-arios	192
6.2.1	Representación en un <i>vector de etiquetas</i>	192
6.2.2	Representación con <i>celdas enlazadas/vector de hijos</i>	194

6.3	La clase NArbol	195
6.3.1	Interfaz pública del tipo <i>NArbol</i>	195
6.3.2	Implementación del tipo <i>NArbol</i>	200
6.3.3	Iteradores sobre un <i>NArbol</i>	203
6.4	B-árboles	205
6.4.1	Introducción y terminología básica	205
6.4.2	Representación de <i>B-árboles</i>	209
6.4.3	La clase <i>BArbol</i>	211
6.4.4	<i>B-árboles</i> y almacenamiento externo	224
6.4.5	Árboles <i>B+</i> y <i>B*</i>	227
6.5	Problemas	230
7	Árboles binarios	233
7.1	Introducción y terminología básica	233
7.1.1	Recorridos	234
7.1.2	Ejemplo: árboles de expresión	235
7.2	Representación de árboles binarios	237
7.2.1	Moviéndose por los nodos del árbol	238
7.2.2	Creación y modificación de árboles	239
7.2.3	Liberar un árbol	240
7.2.4	Listar elementos de un árbol binario	241
7.3	La clase <i>ArbolBin</i>	243
7.3.1	Interfaz pública del tipo <i>ArbolBin</i>	244
7.3.2	Iteradores sobre un árbol binario	248
7.3.3	Implementación del tipo <i>ArbolBin</i>	249
7.4	E/S de árboles binarios. Serialización	254
7.4.1	Árbol binario desde recorridos	254
7.4.2	Preorden con marcas	255
7.5	Problemas	255
8	Árboles binarios de búsqueda	257
8.1	Introducción	257
8.2	ABB simple	258
8.2.1	Búsqueda en un ABB	258
8.2.2	Inserción	259
8.2.3	Borrado	261
8.2.4	Eficiencia	263
8.2.5	Implementación en base a árbol binario	265
8.3	Árboles desplegados	267
8.3.1	Buscar... ¡modifica el árbol!	268
8.3.2	Despliegue	270
8.3.3	Unión y división	273

8.3.4	Búsquedas, inserciones y borrados	274
8.3.5	Eficiencia	278
8.4	Árboles AVL	281
8.4.1	Definición	281
8.4.2	Inserción	283
8.4.3	Borrado	287
8.4.4	Eficiencia	289
8.4.5	Implementación básica con punteros	292
8.4.6	Implementación en base a árbol binario	296
8.5	Árboles Rojo-Negro	298
8.5.1	Nodos externos negros	300
8.5.2	Búsqueda, inserción y borrado	302
8.5.3	Eficiencia de un árbol rojo-negro	321
8.5.4	Relación con árboles 2-4	323
8.6	Problemas	325
9	Colas con prioridad: heaps	329
9.1	Introducción y terminología básica	329
9.2	Heaps: operaciones básicas	330
9.2.1	Operaciones: <i>frente, poner, quitar</i>	331
9.2.2	Implementación sobre un vector-C	332
9.2.3	Cola con prioridad sobre un <i>heap</i>	336
9.2.4	Heaps k-arios	338
9.3	Heaps oblicuos: Mezcla	339
9.3.1	Heaps oblicuos	340
9.3.2	Implementación	342
9.3.3	Eficiencia	345
9.3.4	Mezcla <i>top-down vs bottom-up</i>	349
9.4	Heaps a izquierda	349
9.4.1	Equilibrio de <i>heap a izquierda</i>	349
9.4.2	Camino derecho y número de nodos	351
9.4.3	Mezclar en un <i>heap a izquierda</i>	353
9.4.4	Implementación	353
9.5	Decrementar	355
9.5.1	Decrementar en <i>heap a izquierda</i>	357
9.5.2	Implementación	359
9.5.3	Borrado de un nodo cualquiera	361
9.6	Problemas	362
10	Bosques	363
10.1	Introducción	363

10.2	Heaps binomiales	364
10.2.1	Árbol binomial	364
10.2.2	Definición	367
10.2.3	Operaciones	368
10.2.4	Eficiencia	371
10.2.5	Implementación	372
10.3	Heaps binomiales perezosos	377
10.3.1	Operaciones perezosas	377
10.3.2	El tipo <i>Bosque</i>	378
10.3.3	Mezcla perezosa	382
10.3.4	Implementación	384
10.4	Heaps Fibonacci	390
10.4.1	Árboles Fibonacci	390
10.4.2	Tamaño y número de hijos de un nodo	394
10.4.3	Análisis amortizado	397
10.4.4	Implementación	398
10.4.5	Borrado de un nodo cualquiera	403
10.5	Conjuntos disjuntos	405
10.5.1	Relaciones de equivalencia: ejemplos	405
10.5.2	El bosque: un árbol por cada conjunto	407
10.5.3	Unir/buscar	408
10.5.4	Optimización	410
10.5.5	Implementación	413
10.6	Problemas	416
11	Tablas Hash	419
11.1	Introducción	419
11.1.1	Un ejemplo muy simple	420
11.2	Funciones hash	420
11.2.1	Diseño de funciones hash	422
11.2.2	Hashing de un entero	423
11.2.3	Hashing de una cadena	425
11.2.4	Usos de funciones <i>hash</i>	427
11.2.5	Hashing en el estándar de C++	428
11.3	Resolución de Colisiones	430
11.3.1	Hashing cerrado o direccionamiento abierto	431
11.3.2	Hashing abierto o encadenamiento separado	437
11.3.3	Encadenamiento mezclado	438
11.4	Eficiencia de las tablas hash	441
11.4.1	Factor de carga	442
11.4.2	Comparación de métodos	442
11.4.3	Redimensionamiento y rehashing	445
11.4.4	Tablas hash vs árboles de búsqueda	446

11.5	Implementación	447
11.5.1	Hashing con cursores	447
11.5.2	Hashing con iteradores	455
11.6	Problemas.	464
A	Solución a los ejercicios	467
A.1	Vectores y celdas enlazadas	467
A.2	Eficiencia	471
A.3	Pilas y colas	477
A.4	Listas	485
A.5	Árboles generales	491
A.6	Árboles n-arios	497
A.7	Árboles binarios	505
A.8	Árboles binarios de búsqueda	508
A.9	Colas con prioridad: heaps	518
A.10	Bosques	522
A.11	Tablas Hash	528
B	Tablas	535
B.1	Tabla ASCII	535
B.2	Operadores C++	535
B.3	Palabras reservadas de C89, C99, C11, C++ y C++11	538
	Bibliografía	541
	Referencias electrónicas	544
	Índice alfabético	545