

UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

CC4102 - Diseño y Analisis de Algoritmos

Tarea 1: Diccionarios1

Cristian Carreño Medina
Diego Chávez Escobar
Prof. Jeremy Barbay; Aux. Mauricio Quezada

Índice general

1. Presentación	1
1.1. Introducción	1
2. Experimentacion	2
2.1. Marco Teorico	2
2.1.1. Estructuras	2
2.2. Diseño Experimental	3
3. Resultados y Analisis	4
3.1. Resultados	4
3.1.1. Arbol AVL	4
3.1.2. Arbol Rojo-Negro	5
3.1.3. Arbol 2-3	5
3.1.4. Arbol B	6
3.1.5. Arbol AVL	6
3.1.6. Arbol Rojo-Negro	7
3.1.7. Arbol 2-3	7
3.1.8. Arbol B	8
3.1.9. Arbol AVL	8

3.1.10. Arbol Rojo-Negro	9
3.1.11. Arbol 2-3	9
3.1.12. Arbol B	10
3.1.13. Arbol AVL	10
3.1.14. Arbol Rojo-Negro	11
3.1.15. Arbol 2-3	11
3.1.16. Arbol B	12
3.2. Analisis de Resultados	12
4. Conclusiones	14
4.1. Conclusiones	14

Capítulo 1

Presentación

1.1. Introducción

En el presente informe se detallaran los resultados en terminos de eficiencia de cuatro estructuras de datos, se pondran a prueba las funciones de insercion, busqueda y eliminacion de datos, en las siguientes estructuras:

1. Arbol AVL
2. Arbol Rojo-Negro
3. Arbol 2-3
4. Arbol B

Capítulo 2

Experimentacion

2.1. Marco Teorico

2.1.1. Estructuras

Arbol AVL

Estos Arboles están siempre equilibrados de tal modo que para todos los nodos, la altura de la rama izquierda no difiere en más de una unidad de la altura de la rama derecha o viceversa. Gracias a esta forma de equilibrio (o balanceo), la complejidad de una búsqueda en uno de estos árboles se mantiene siempre en orden de complejidad $O(\log n)$. El factor de equilibrio puede ser almacenado directamente en cada nodo o ser computado a partir de las alturas de los subárboles.

Arbol Rojo-Negro

Un árbol rojo-negro es un árbol binario de búsqueda en el que cada nodo tiene un atributo de color cuyo valor es o bien rojo o bien negro, las hojas poseen valor nulos, los hijos de todo nodo rojo es negro, y el valor de nodos negros en el camino es constante independiente del camino, con esto se logra un arbol aproximadamente equilibrado.

Arbol 2-3

Un árbol 2-3 permite que un nodo tenga dos o tres hijos ademas que todas las hojas han de estar al mismo nivel. Esta característica le permite conservar el balanceo tras insertar o borrar elementos, por lo que el algoritmo de búsqueda es casi tan rápido como en un árbol

de búsqueda de altura mínima. Por otro lado, es mucho más fácil de mantenerlo.

Arbol B

Los árboles-B poseen un numero variable de nodos hijos con un maximo predifnido. Cuando se inserta o se elimina un dato de la estructura, la cantidad de nodos hijo varía dentro de un nodo. Para que siga manteniéndose el número de nodos dentro del rango predefinido, los nodos internos se juntan o se parten. Dado que se permite un rango variable de nodos hijo, los árboles-B no necesitan rebalancearse tan frecuentemente como los árboles binarios de búsqueda auto-balanceables, pero por otro lado pueden desperdiciar memoria, porque los nodos no permanecen totalmente ocupados. Los límites superior e inferior en el número de nodos hijo son definidos para cada implementación en particular.

2.2. Diseño Experimental

Para reazlizar este experimento, se comenzara por crear la secuencia de insercion, la cual sera una secuencia semi-ordenada de elementos y al azar de elementos segun los casos a estudiar, luego de esto se crea una secuencia de borrado al azar y semi-ordenada que contendra elementos que seran insertos en el arreglo.

Con estos arreglos se podra testear las secuencias de insercion y borrado que terminaban una vez se acaban las secuencias creadas, marcado su tiempo de ejecucion.

Del mismo modo para los test que requieren de una secuencia de busqueda se crea una secuencia de busqueda de $k/2$ veces la secuencia de insercion, de este modo se tiene un arreglo que posee los elementos a ser buscados en los arboles.

Luego de esto se testean la ejecucion de las instrucciones de insercion, busqueda y borrado y se mide los tiempos de ejecucion para cada estructura de datos.

Capítulo 3

Resultados y Analisis

3.1. Resultados

1. Los elementos de insercion y busqueda son escogidos al azar de universo

3.1.1. Arbol AVL

Prueba k=3

- a) $(i^k d^k i^k)^n = 169 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 74 \text{ ms}$
- c) $(d^k i^k d^k)^n = 66 \text{ ms}$

Prueba k=5

- a) $(i^k d^k i^k)^n = 118 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 139 \text{ ms}$
- c) $(d^k i^k d^k)^n = 67 \text{ ms}$

Prueba k=7

- a) $(i^k d^k i^k)^n = 168 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 187 \text{ ms}$
- c) $(d^k i^k d^k)^n = 69 \text{ ms}$

3.1.2. Arbol Rojo-Negro

Prueba k=3

- a) $(i^k d^k i^k)^n = 544$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 136$ ms
- c) $(d^k i^k d^k)^n = 150$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 93$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 68$ ms
- c) $(d^k i^k d^k)^n = 71$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 101$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 97$ ms
- c) $(d^k i^k d^k)^n = 72$ ms

3.1.3. Arbol 2-3

Prueba k=3

- a) $(i^k d^k i^k)^n = 1433$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 666$ ms
- c) $(d^k i^k d^k)^n = 253$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 260$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 375$ ms
- c) $(d^k i^k d^k)^n = 222$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 226$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 433$ ms
- c) $(d^k i^k d^k)^n = 209$ ms

3.1.4. Arbol B

Prueba k=3

- a) $(i^k d^k i^k)^n = 414$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 115$ ms
- c) $(d^k i^k d^k)^n = 45$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 119$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 43$ ms
- c) $(d^k i^k d^k)^n = 36$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 132$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 52$ ms
- c) $(d^k i^k d^k)^n = 34$ ms

2. Los elementos de insertar y borrar se encuentran semi-ordenados

3.1.5. Arbol AVL

Prueba k=3

- a) $(i^k d^k i^k)^n = 37$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 21$ ms
- c) $(d^k i^k d^k)^n = 17$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 143$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 1349$ ms
- c) $(d^k i^k d^k)^n = 23$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 148$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 2649$ ms
- c) $(d^k i^k d^k)^n = 45$ ms

3.1.6. Arbol Rojo-Negro

Prueba k=3

- a) $(i^k d^k i^k)^n = 50$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 41$ ms
- c) $(d^k i^k d^k)^n = 40$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 34$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 34$ ms
- c) $(d^k i^k d^k)^n = 78$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 36$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 59$ ms
- c) $(d^k i^k d^k)^n = 39$ ms

3.1.7. Arbol 2-3

Prueba k=3

- a) $(i^k d^k i^k)^n = 134$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 169$ ms
- c) $(d^k i^k d^k)^n = 119$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 145$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 292$ ms
- c) $(d^k i^k d^k)^n = 128$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 191$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 483$ ms
- c) $(d^k i^k d^k)^n = 138$ ms

3.1.8. Arbol B

Prueba k=3

- a) $(i^k d^k i^k)^n = 33782$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 874$ ms
- c) $(d^k i^k d^k)^n = 85135$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 35996$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 1279$ ms
- c) $(d^k i^k d^k)^n = 92453$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 34420$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 1773$ ms
- c) $(d^k i^k d^k)^n = 95593$ ms

3. Solo elementos a insertar estan semi-ordenados

3.1.9. Arbol AVL

Prueba k=3

- a) $(i^k d^k i^k)^n = 144$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 152$ ms
- c) $(d^k i^k d^k)^n = 55$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 142$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 335$ ms
- c) $(d^k i^k d^k)^n = 53$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 144$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 272$ ms
- c) $(d^k i^k d^k)^n = 40$ ms

3.1.10. Arbol Rojo-Negro

Prueba k=3

- a) $(i^k d^k i^k)^n = 34 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 16 \text{ ms}$
- c) $(d^k i^k d^k)^n = 27 \text{ ms}$

Prueba k=5

- a) $(i^k d^k i^k)^n = 34 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 27 \text{ ms}$
- c) $(d^k i^k d^k)^n = 26 \text{ ms}$

Prueba k=7

- a) $(i^k d^k i^k)^n = 34 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 41 \text{ ms}$
- c) $(d^k i^k d^k)^n = 25 \text{ ms}$

3.1.11. Arbol 2-3

Prueba k=3

- a) $(i^k d^k i^k)^n = 168 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 203 \text{ ms}$
- c) $(d^k i^k d^k)^n = 174 \text{ ms}$

Prueba k=5

- a) $(i^k d^k i^k)^n = 166 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 343 \text{ ms}$
- c) $(d^k i^k d^k)^n = 177 \text{ ms}$

Prueba k=7

- a) $(i^k d^k i^k)^n = 164 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 478 \text{ ms}$
- c) $(d^k i^k d^k)^n = 175 \text{ ms}$

3.1.12. Arbol B

Prueba k=3

- a) $(i^k d^k i^k)^n = 62 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 26 \text{ ms}$
- c) $(d^k i^k d^k)^n = 26 \text{ ms}$

Prueba k=5

- a) $(i^k d^k i^k)^n = 81 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 47 \text{ ms}$
- c) $(d^k i^k d^k)^n = 26 \text{ ms}$

Prueba k=7

- a) $(i^k d^k i^k)^n = 94 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 66 \text{ ms}$
- c) $(d^k i^k d^k)^n = 30 \text{ ms}$

4. Solo los elementos a borrar estan semi ordenados

3.1.13. Arbol AVL

Prueba k=3

- a) $(i^k d^k i^k)^n = 38 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 21 \text{ ms}$
- c) $(d^k i^k d^k)^n = 18 \text{ ms}$

Prueba k=5

- a) $(i^k d^k i^k)^n = 139 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 561 \text{ ms}$
- c) $(d^k i^k d^k)^n = 21 \text{ ms}$

Prueba k=7

- a) $(i^k d^k i^k)^n = 148 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 1226 \text{ ms}$
- c) $(d^k i^k d^k)^n = 23 \text{ ms}$

3.1.14. Arbol Rojo-Negro

Prueba k=3

- a) $(i^k d^k i^k)^n = 35 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 16 \text{ ms}$
- c) $(d^k i^k d^k)^n = 26 \text{ ms}$

Prueba k=5

- a) $(i^k d^k i^k)^n = 35 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 26 \text{ ms}$
- c) $(d^k i^k d^k)^n = 21 \text{ ms}$

Prueba k=7

- a) $(i^k d^k i^k)^n = 34 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 40 \text{ ms}$
- c) $(d^k i^k d^k)^n = 26 \text{ ms}$

3.1.15. Arbol 2-3

Prueba k=3

- a) $(i^k d^k i^k)^n = 132 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 168 \text{ ms}$
- c) $(d^k i^k d^k)^n = 119 \text{ ms}$

Prueba k=5

- a) $(i^k d^k i^k)^n = 146 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 288 \text{ ms}$
- c) $(d^k i^k d^k)^n = 127 \text{ ms}$

Prueba k=7

- a) $(i^k d^k i^k)^n = 143 \text{ ms}$
- b) $f^{kn}(d^k i^k d^k)^n = 389 \text{ ms}$
- c) $(d^k i^k d^k)^n = 127 \text{ ms}$

3.1.16. Arbol B

Prueba k=3

- a) $(i^k d^k i^k)^n = 36949$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 747$ ms
- c) $(d^k i^k d^k)^n = 84675$ ms

Prueba k=5

- a) $(i^k d^k i^k)^n = 36794$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 1248$ ms
- c) $(d^k i^k d^k)^n = 95008$ ms

Prueba k=7

- a) $(i^k d^k i^k)^n = 38139$ ms
- b) $f^{kn}(d^k i^k d^k)^n = 1778$ ms
- c) $(d^k i^k d^k)^n = 97539$ ms

3.2. Analisis de Resultados

De los resultados obtenidos se puede determinar que:

1. Los elementos de insercion y borrado son escogidos al azar
Los resultados del arbol Rojo-Negro son excelentes incluso aumentando el numero de datos, y que al aumntar los datos mantiene tiempos similares, en cambio el arbol 2-3 y AVL poseen marcas medias y similares, salvo que para el borrado el arbol AVL posee un desempeño mejor. Por su parte el arbol B posee resultados muy buenos y mejores que el arbol 2-3 y AVL, pero no comparables con el Rojo-Negro. En este experimentos que todas las pruebas realizadas para lso arboles AVL, Rojo-negro y B, poseen marcas similares, mientras que el arbol 2-3 posee ligeramente mayor. Esto se puede explicar debido a el autobalance que poseen estos arboles provocando marcas similares.
2. Los elementos de insercion y borrado se encuentran semi-ordenados

En estos resultados destaca en tiempos el arbol Rojo-negro, mientras el arbol AVL posee tiempos mayores de busqueda. Por su parte el arbol 2-3 no posee grnades cambios versus el experimento anterior. En cambio el arbol B tiene un resultado critico en lo que refiere a insercion y borrado.

3. Solo los elementos a insertar se encuentran semi-ordenados

Los resultados del arbol Rojo-Negro son excelentes incluso aumentando el numero de datos, y que al aumentar los datos mantiene tiempos similares, en cambio el arbol 2-3 y AVL poseen marcas medias y similares, salvo que para el borrado el arbol AVL posee un desempeño mejor. Por su parte el arbol B posee resultados muy buenos y mejores que el arbol 2-3 y AVL, pero no comparables con el Rojo-Negro.

4. Solo los elementos a borrar se encuentran semi-ordenados

El arbol AVL siguiendo su linea posee desempeño estandar, al igual que el arbol 2-3, el cual solo supera al AVL en las busquedas. Por su parte el Arbol Rojo-Negro vuelve a demostrar un desempeño excelente en comparacion a los otros arboles. En cambio, el arbol B posee una desempeño horrible en comparacion a los otros arboles.

Capítulo 4

Conclusiones

4.1. Conclusiones

Del presente informe de experimentacion se puede concluir que los tiempos de busqueda, insercion y borrado, ademas de la estructura a utilizar depende netamente del caso de uso al cual sera sometido, ya que se pudo observar tiempos muy variables para estruturas sometidas a pruebas distintas, ya sea el caso de ordenar arreglos al azar o semi-ordenados.

De esto podemos concluir que los arboles AVL no poseen la mejor perfomace, pero entregan tiempos estables independientes de los casos, por su parte el arbol 2-3 posee tiempos ligeramente mejores que el AVL pero es superado por este en ela busqueda.

Por su parte el arbol Rojo-Negro posee marca promedio para arreglos al azar, pero basta que el arreglo de insercion o busqueda este semi-ordenado para que este obtenga resutados realmente buenos. En cambio el arbol B posee muy buenas marcas cuando el arreglo de borrado es al azar, pero basta que este sea semi-ordenado para que este presente tiempos mediocres e inacceptables.