

ESTRUCTURAS DE DATOS

TRABAJO PRÁCTICO N ° 7
Árboles Binarios

Departamento de Ciencias e Ingeniería de la Computación - U.N.S.
Primer cuatrimestre de 2019

Bibliografía:

- [GT] Michael Goodrich & Roberto Tamassia. *Data Structures and Algorithms in Java. Fourth Edition* . John Wiley and Sons. 2006.
- [W] Mark A. Weiss. *Estructuras de datos* . Addison-Wesley Iberoamericana España, S.A. 2000.
- [W12] Mark A. Weiss. *Data Structures and Algorithm Anlysis in Java*. Third Edition. Addison-Wesley Pearson Education, Inc. 2012.
-

Ejercicio 1:

- Defina recursivamente el concepto de árbol binario.
- ¿Qué diferencia hay entre los árboles utilizados hasta ahora y los árboles binarios?
- ¿Es posible utilizar las estructuras de datos anteriores para representar árboles binarios? ¿Existe alguna otra estructura nueva que posee ventajas sobre las anteriores?
- Implemente los recorridos: orden-previo, orden-posterior, orden-simétrico y por niveles, para un árbol binario.
- Redefina el método clone() para árboles binarios que permita retornar una copia del árbol receptor del mensaje.

Ejercicio 2:

Dados la altura H y el recorrido en orden previo de un árbol binario A perfecto, ¿es posible reconstruir el árbol original? Nota: Un árbol binario perfecto es un árbol binario propio en el cual todas las hojas están a la misma profundidad. Un árbol binario es propio si todos los nodos excepto las hojas tienen dos hijos.

Ejercicio 3:

- Defina todas las estructuras de datos necesarias para implementar árboles binarios (AB) utilizando referencias. El tipo del rótulo es de tipo genérico E.
- Escriba una función que cuenta la cantidad de nodos de un AB. Resuelva este ejercicio de dos maneras: (i) La función recibe un AB y lo manipula sólo a través de las operaciones del TDA AB y (ii) agregue una operación a la clase AB y manipule el árbol a través de la estructura de nodos subyacentes.
- Escriba un método que dado un AB devuelva un "espejo" del mismo. Realice las mismas consideraciones que en el inciso (c).
- Escriba un método tal que dados dos nodos N1 y N2 de un AB A indique si existe un camino entre N1 y N2.
- Modifique el método anterior para que devuelva una lista con el camino hallado.
- Escriba un método que indique la profundidad de un nodo N de un AB A, e indique el camino que la justifica.
- Escriba un método para hallar la altura de un nodo N de un AB A e indique el camino que la justifica.

Ejercicio 4:

Realice los siguientes incisos utilizando las operaciones del TDA cuando lo crea conveniente, o de lo contrario trabaje directamente sobre la estructura de datos. Elija la implementación de árboles que crea más apropiada en cada caso.

- a) Escriba un método que indique si un árbol binario es propio [GT], es decir todo nodo tiene o bien dos hijos, o bien ningún hijo.
- b) Escriba un método que muestre todos los nodos que son padres de una hoja, indicando el nivel en que se encuentran.
- c) Escriba un método tal que dados dos árboles A y A1 determine si A1 es un subárbol de A.
- d) Escriba un método que verifique si dos árboles son iguales.

Ejercicio 5:

- a) Dado un árbol binario A que contiene una expresión aritmética. Escriba un método que imprima en pantalla la expresión en notación prefija, sufija, e infija con los paréntesis indispensables.
- b) Escriba un método que cree el AB de expresión correspondiente a partir del listado de una expresión aritmética que se ingresa por teclado y está formada por dígitos, y operadores de suma, resta, producto y división.
- c) Escriba un método para evaluar una expresión aritmética representada como un AB de expresión.

Ejercicio 06:

- a) Defina un árbol binario de búsqueda (ABB)
- b) ¿Cuál es el objetivo de este tipo de estructuras de datos?
- c) ¿Existe alguna forma de recorrer el árbol tal que obtenga todos los elementos ordenados de menor a mayor? ¿Y si desea obtenerlos ordenados en forma decreciente?
- d) Dibuje todos los ABB que se pueden obtener para representar el conjunto $C = \{2, 4, 8, 10\}$. ¿Cuál de los ABB anteriores es mejor en función del objetivo expuesto en el inciso b)?
- e) Implemente un TDA correspondencia utilizando un ABB. Implemente además la operación MIN que retorna la mínima clave del ABB.
- f) Implemente un TDA diccionario utilizando un ABB.
- g) Compare los tiempos de ejecución de la operación find (Pertenece) implementada con una tabla de dispersión y con un ABB en el mejor y peor caso.
- h) Muestre el ABB resultante de ingresar los elementos (números enteros) 7, 2, 9, 0, 5, 6, 8, 1, en ese orden con la operación insertar, partiendo de un árbol vacío. Luego muestre el árbol resultante de eliminar los enteros 7 y 2, en ese orden y en orden inverso (ie, 2 y 7). El árbol resultante ¿es el mismo?

Ejercicio 07:

- a) Realice un procedimiento que elimine los nodos de un ABB que no tienen rótulo R y que estén en el nivel 3.
- b) Realice un procedimiento para verificar si un árbol binario A es o no un ABB según la definición 11)a).
- c) Sea B un ABB y R1 y R2 rótulos de B. Realizar un procedimiento para eliminar de B el ancestro común de R1 y R2 más cercano.

Ejercicio 08:

- a) Escriba en Java todas las estructuras de datos necesarias para implementar un mapeo genérico de claves de tipo K y valores de tipo V usando un árbol binario de búsqueda. Discuta variantes a la implementación cuando utiliza un nodo parametrizado con un tipo E y otra cuando utiliza un nodo parametrizado con dos tipos K y V.
- b) Implemente las operaciones de la interfaz Map dada en la teoría utilizando la estructura programada en (a). Utilice como guía la explicación de la teoría modificando el código y proponiendo extensiones del mismo cuando lo considerara adecuado.
- c) Realice un análisis del tiempo de ejecución de su implementación cuando (1) el árbol de entrada está lleno y (2) el árbol de entrada está desbalanceado.

Ejercicio 09:

- a) Dados dos mapeos M1 y M2, M1 implementado con un árbol binario de búsqueda y M2 implementado con un tabla de dispersión abierta, se quiere determinar si todos los elementos de M1 están contenidos en M2. Analice variaciones de la solución si el método solución es:
- b) agregado como un método de la clase que implementa el mapeo M1 (esta operación sólo recibe un mapeo M2)
- c) agregado como un método de la clase que implementa el mapeo M2 (esta operación sólo recibe un mapeo M1)
- d) implementado por una clase de servicio que no tiene acceso ni a la estructura de datos de M1 ni a la de M2 (esta operación recibe un mapeo M1 y otro M2).

Indique el orden del tiempo de ejecución de las soluciones implementadas. Justifique apropiadamente.

Ejercicio 10:

Sea el siguiente problema: Dado un mapeo M de claves de tipo K y valores de tipo V implementado con un árbol binario de búsqueda y un valor A, implemente un método que retorne otro mapeo M2 con aquellas entradas de M que fueran mayores a A. Caracterice el orden del tiempo de ejecución de la solución propuesta.

- a) Resuelva el problema con un método que pertenece a la clase Mapeo que implementa la interfaz Map.
- b) Resuelva el problema con un método que no pertenece a la clase Mapeo que implementa la interfaz Map.

Ejercicio 11:

- a) Escriba en Java todas las estructuras de datos necesarias para implementar un diccionario de claves de tipo K y valores de tipo V usando un árbol binario de búsqueda. Discuta variantes a la implementación cuando (1) permite claves repetidas en los nodos del árbol binario de búsqueda, y (2) cuando el árbol binario de búsqueda no tiene nodos con claves repetidas.
- b) Implemente la interfaz Dictionary dada en la teoría usando la estructura de datos propuesta en (a).

- c) Dado un diccionario D de claves de tipo K y valores de tipo V implementado con un árbol binario de búsqueda, implemente un método que retorne un mapeo M donde para cada clave k de D, el mapeo M retorna la cantidad de ancestros que tiene el nodo de D que tiene como clave a k. Caracterice el orden del tiempo de ejecución de la solución propuesta.

Ejercicio 12 (Extraído de un examen final):

Resuelva el siguiente problema: Dados dos ABB A y B se desea construir un tercer ABB C donde los elementos de C son aquellos que se hallan en A y no en B. Asuma que la solución al problema es un método de la clase ABB y que A recibe el mensaje. Estime el orden del tiempo de ejecución de su solución justificando apropiadamente.

Ejercicio 13 (Extraído de un examen final):

Resuelva el siguiente problema: Dado un conjunto S implementado con un ABB y dados tres enteros i, j y k, se desean obtener los elementos de S ubicados en las posiciones i, j y k de S de acuerdo a su ordenación por \leq . Nota: La ordenación de S inducida por \leq es una lista $[a_1, a_2, \dots, a_{i-1}, a_i, a_{i+1}, \dots, a_n]$ tal que $a_1 \leq a_2 \leq \dots \leq a_{i-1} \leq a_i \leq a_{i+1} \leq \dots \leq a_n$. De el tiempo de ejecución de su solución justificando apropiadamente. Explique si el método (o los métodos) a programar son parte o no de la clase ABB y por qué lo decidió de esa manera.

Ejercicio 14 (Extraído de un examen final):

Programe una operación llamada *Ejercicio14* que forma parte de la clase árbol binario de búsqueda (y por lo tanto tiene acceso a la estructura interna del mismo) que recibe un rótulo de nodo y debe retornar el rótulo del predecesor inorden de dicho nodo. Programe todas las operaciones auxiliares de árbol binario de búsqueda que use. Calcule el orden del tiempo de ejecución de su solución justificando apropiadamente.