

Comenzado el	domingo, 12 de noviembre de 2023, 18:21
Estado	Finalizado
Finalizado en	domingo, 12 de noviembre de 2023, 18:29
Tiempo empleado	8 minutos 10 segundos
Calificación	9 de 10 (85%)

Pregunta **1**

Correcta

Se puntúa 1 sobre 1

Sabemos que en un árbol binario (sea o no de búsqueda) un nodo puede tener a lo sumo dos hijos. Por otra parte, un nodo tendrá un único nodo padre (salvo la raíz, que no tiene padre) ¿Qué parte de la definición de árbol binario es la que impide que un nodo pueda tener más de un padre?

Seleccione una:

- ☒ a. La sección en la que se afirma que los dos subconjuntos que dependen del nodo raíz son conjuntos separados o mutuamente excluyentes. ✓
- ☐ b. La sección en la que se afirma que el árbol puede estar vacío o puede contener un nodo designado como la raíz del árbol.
- ☐ c. La sección en la que se afirma que del nodo raíz se desprenden dos subconjuntos.
- ☐ d. La sección en la que se afirma que los dos subconjuntos que dependen del nodo raíz pueden estar vacíos.

Pregunta **2**

Correcta

Se puntúa 1 sobre 1

¿Cuál de las siguientes es CIERTA respecto de los árboles binarios de búsqueda?

Seleccione una:

- ☐ a. Siempre tienen altura mínima.
- ☐ b. Siempre tienen altura máxima.
- ☐ c. Si el árbol tiene n nodos, el tiempo de búsqueda en el peor caso es $O(n \cdot \log(n))$ si la altura del árbol h es $O(\log(n))$.
- ☒ d. Si el árbol tiene n nodos, el tiempo de búsqueda en el peor caso es $O(\log(n))$ si la altura del árbol h es $O(\log(n))$. ✓

Pregunta **3**

Correcta

Se puntúa 1 sobre 1

¿Cuántas comparaciones **por nivel** se realizan en un árbol de búsqueda de n nodos cuando se busca una clave? (no importa si la búsqueda llega o no al último nivel del árbol: sólo queremos saber cuántas comparaciones se hacen en aquellos niveles hasta los que llegue la búsqueda)

Seleccione una:

- ☐ a. Comparaciones por nivel: n
- ☐ b. Comparaciones por nivel: n^2
- ☒ c. Comparaciones por nivel: 1 (una) ✓
- ☐ d. Comparaciones por nivel: $\log(n)$

Pregunta **4**

Parcialmente correcta

Se puntúa 1 sobre 1

Considere el método `contains()` que busca un objeto x en un árbol de búsqueda de n nodos, según se vió en las fichas de clase de la asignatura:

```
def contains(self, x):
    if x is None:
        return False

    p = self.root
    while p is not None:
        if x == p.info:
            return True

        elif x < p.info:
            p = p.left
        else:
            p = p.right

    return False
```

Si el algoritmo de inserción mediante el cual se ha creado el árbol es el algoritmo simple visto en la ficha y en clases, sin control de altura o balance, ¿cuál de las siguientes afirmaciones es correcta en relación al algoritmo implementado en el método `contains()`?

Seleccione una o más de una:

- ☐ a. El tiempo de ejecución del algoritmo mostrado es $O(\log(n))$ en el peor caso.
- ☒ b. El tiempo de ejecución del algoritmo mostrado es $O(n)$ en el peor caso. ✓
- ☐ c. El tiempo de ejecución del algoritmo mostrado es $O(\log(n))$ en el peor caso, pero sólo si el árbol tiene altura $h = O(\log(n)) = \text{mínima}$.
- ☒ d. El tiempo de ejecución del algoritmo mostrado es $O(1)$ en el peor caso. ✗

Pregunta 5

Incorrecta

Se puntúa 0 sobre 1

Suponga que desconoce la forma gráfica de un árbol binario (**que no es necesariamente de búsqueda**) con el que está trabajando, pero se le da la posibilidad de conocer una o dos secuencias de recorrido de ese árbol, en distintos órdenes. *¿Con cuáles de las siguientes secuencias de recorrido se garantiza que podrá volver a dibujar el árbol original, sin ambigüedades ni errores?* (Piense un poco para responder, y en todo caso, haga algunos intentos dibujando un árbol y tratando de reconstruirlo a partir de sus secuencias de recorrido) **(Observación: más de una respuesta puede ser correcta. Marque todas las que considere válidas). Recuerde: se supone que el árbol original es binario, pero NO NECESARIAMENTE de búsqueda.**

Seleccione una o más de una:

- ☒ a. Podrá volver a dibujar el árbol si conoce la secuencia de recorrido en *preorden* y la secuencia de recorrido en *postorden*. ✖
- ☐ b. Podrá volver a dibujar el árbol si sólo conoce la secuencia de recorrido en *entre orden*.
- ☒ c. Podrá volver a dibujar el árbol si conoce la secuencia de recorrido en *postorden* y la secuencia de recorrido en *entreorden*. ✔
- ☐ d. Podrá volver a dibujar el árbol si conoce la secuencia de recorrido en *preorden* y la secuencia de recorrido en *entreorden*.

Pregunta 6

Correcta

Se puntúa 1 sobre 1

¿En qué condiciones el **recorrido en preorden** de un **árbol binario de búsqueda** visitará los nodos del árbol en secuencia ordenada de menor a mayor?

Seleccione una:

- ☐ a. Sólo si el árbol es una diagonal con todos los hijos derechos nulos.
- ☒ b. Sólo si el árbol es una diagonal con todos los hijos izquierdos nulos. ✔
- ☐ c. Nunca.
- ☐ d. Siempre.

Pregunta 7

Correcta

Se puntúa 1 sobre 1

¿Qué es lo peor que puede pasar con un árbol de búsqueda de n nodos si no se lo mantiene equilibrado (o balanceado) (o sea, intuitivamente, si los n nodos se distribuyen en más de $\log(n)$ niveles)?

Seleccione una:

- ☐ a. Que una búsqueda obligue a $O(\log(n))$ comparaciones.
- ☐ b. Que una búsqueda obligue a $O(n^2)$ comparaciones.
- ☒ c. Que una búsqueda obligue a $O(n)$ comparaciones. ✔
- ☐ d. Que una búsqueda obligue a $O(2^n)$ comparaciones.

Pregunta 8



Correcta

Se puntúa 1 sobre 1

Típicamente (y por diversas razones) en un árbol binario de búsqueda no se acepta la inserción de claves o valores repetidos. Pero existen estrategias simples que pueden usarse para permitir esa repetición de claves. Una de ellas es la de relajar la regla de inserción de forma que los menores sigan yendo a la izquierda, pero ahora los **mayores o iguales** vayan a la derecha.

¿Cuáles de las siguientes afirmaciones son ciertas si en un árbol binario de búsqueda se cambia la regla de inserción para aplicar la regla relajada citada más arriba? (Más de una respuesta puede ser cierta, por lo que marque todas las que considere correctas).

Seleccione una o más de una:

- ☒ a. Si solo se insertan claves que sean iguales a la que contiene la raíz, el árbol tendrá estructura de una diagonal con todos los  enlaces izquierdos nulos.
- ☐ b. Si solo se insertan claves que sean iguales a la que contiene la raíz, el árbol tendrá estructura de una diagonal con todos los enlaces derechos nulos.
- ☐ c. Dos claves iguales quedarán siempre en el árbol una como hija derecha e inmediata de la otra.
- ☒ d. Dos o más claves iguales podrán quedar separadas por dos o más niveles de distancia. 


Pregunta 9

Correcta

Se puntúa 1 sobre 1

¿Cuál es la razón por la que es recomendable usar una pila como soporte al recorrer un árbol binario (y por lo tanto, cuál es la razón por la que es recomendable usar recursión para ese recorrido)?

Seleccione una:

- ☐ a. En realidad, es mejor usa una cola en lugar de una pila. Una cola permite almacenar en orden FIFO las direcciones de los nodos que se acaban de visitar en el recorrido, y cuando se necesite regresar a ese nodo luego de procesar alguno de sus dos hijos, su dirección quedará en el frente de la cola y será accesible en forma directa.
- ☐ b. No hay ninguna razón para usar una pila o para usar recursión. Un ciclo y una referencia de persecución serían siempre suficientes para hacer el recorrido completo.
- ☒ c. Porque una pila permite almacenar en orden LIFO las direcciones de los nodos que se acaban de visitar en el recorrido, y  cuando se necesite regresar a ese nodo luego de procesar alguno de sus dos hijos, su dirección quedará en el tope de la pila y será accesible en forma directa.
- ☐ d. No es necesario ni usar una pila ni usar recursión. El árbol puede recorrerse simplemente con un ciclo que comience en la raíz y luego se desplace vuelta tras vuelta hacia el hijo izquierdo, hasta llegar a un hijo izquierdo nulo, momento en el cual el ciclo debe detenerse y el árbol queda completamente recorrido.

Pregunta **10**

Correcta

Se puntúa 1 sobre 1

Sabemos que para que un árbol binario de búsqueda favorezca búsquedas rápidas, el árbol debería tener cierta estructura a la que hemos designado como balance o equilibrio. No hemos dado una definición formal y detalla del concepto de balance o equilibrio (ya que eso escapa al alcance de esta asignatura) en un árbol binario: solo hemos tomado una idea intuitiva y muy conceptual.

En base a esa idea intuitiva, ¿cuál de las siguientes describe **conceptualmente** mejor lo que se entiende por un árbol balanceado o equilibrado?

Seleccione una:

- ☐ a. Intuitivamente, un árbol binario está balanceado si tiene forma de zig-zag, con no más de un nodo en cada nivel del árbol.
- ☐ b. Intuitivamente, un árbol binario está balanceado si su altura es la máxima posible (o se acerca mucho a esa máxima) para la cantidad de nodos que tiene el árbol.
- ☐ c. Intuitivamente, un árbol binario está balanceado si tiene forma de diagonal (hacia la derecha o hacia la izquierda).
- ☒ d. Intuitivamente, un árbol binario está balanceado si su altura es la mínima posible (o se acerca mucho a esa mínima) para la cantidad de nodos que tiene el árbol. ✓

◀ [Materiales Adicionales para la Ficha 29](#)

Ir a...



[Trabajo Práctico Recuperatorio: Sistema para una Agencia de Noticias](#) ▶