

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMIA

Laboratorio 4

Segundo Semestre de 2019

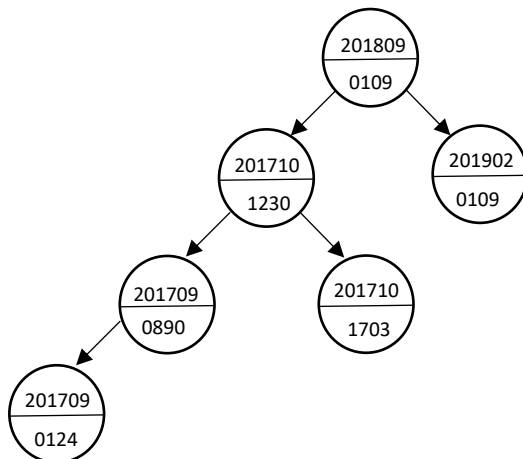
Indicaciones generales:

- Duración: 2h 50 min.
 - Al inicio de cada programa, el alumno deberá incluir, a modo de comentario, la estrategia que utilizará para resolver el problema. De no incluirse dicho comentario, el alumno perderá el derecho a reclamo en esa pregunta.
 - Si la implementación es significativamente diferente a la estrategia indicada o no la incluye, la pregunta no será corregida.
 - Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 50% del puntaje asignado a dicha pregunta.
 - Debe utilizar comentarios para explicar la lógica seguida en el programa elaborado.
 - El orden será parte de la evaluación.
 - Su trabajo deberá ser subido a PAIDEIA en el espacio indicado por los jefes de práctica.
-

Esta evaluación presenta 3 preguntas electivas de las cuales Ud. debe elegir solo 2 de ellas. Por tal motivo debe adjuntar un archivo de texto indicando claramente que preguntas ha decidido desarrollar.

Pregunta 1 (10 puntos)

Una empresa logística desea dividir su almacén principal en varios almacenes independientes. En la actualidad los productos del almacén principal están gestionados utilizando un árbol de búsqueda binaria, pero debido a que su ingreso no ha tenido orden en particular, el árbol no tiene una estructura fácil de dividir. Así mismo se conoce que la información registrada en cada nodo tiene 2 partes: La primera muestra el año y mes en que fue registrado el producto en la empresa, y la segunda parte es un número secuencial menor que va de 0 a 9999, el cual no se repite para un mismo año y mes. Por tal motivo el árbol de búsqueda binaria esta ordenada considerando la combinación de ambas partes como se muestra a continuación:



- a) Implemente una función que permita cargar los datos del árbol de búsqueda binaria, desde el teclado. El número de datos que contiene el árbol o se van a ingresar no es conocido (0.5 puntos).
- b) Un especialista en algoritmia, ha sugerido utilizar para el almacén un árbol binario balanceado, ya que su distribución de los nodos facilita la división de la estructura en grupos similares. Por tal motivo desarrolle una función que reciba como parámetro un árbol binario de búsqueda cualquiera y genere un árbol balanceado binario de búsqueda. Para realizar este proceso **no debe usar recursión**. Además para esta operación puede utilizar cualquier estructura como apoyo (4.0 puntos).
- c) Para verificar que los resultados son correctos imprima el árbol balanceado utilizando el recorrido **preorden recursivo** (0.5 puntos).

Parte Electiva:

- d) Desarrolle una función que imprima los nodos del árbol balanceado resultante, utilizando un recorrido **iterativo** visto en clase, tal que permita reconstruir dicho árbol de búsqueda binaria, a partir de lo mostrado o impreso (5 puntos).

Pregunta 2 (10 puntos)

Existe un servicio de bus turístico que tiene una ruta con n paradas. Cada parada está identificada con un número de 1 al n. La ruta cubre diversos atractivos turísticos de la ciudad, tanto desde la parada 1 a la n, como el recorrido inverso desde la parada n a la 1. Si un turista está en la parada p y quiere ir a la parada q, y todas las paradas están disponibles, entonces simplemente tendría que tomar el bus en la parada p y dirigirse a la parada q, usando el recorrido apropiado. Sin embargo, cada día hay paradas que no están disponibles en los recorridos. Cuando una parada no está disponible, el bus simplemente no se detiene en esa parada. Una representación de este escenario es la siguiente:

1	0	0	0	1
0	1	1	1	1

La primera fila indica la disponibilidad de las paradas en la ida. La segunda fila indica la disponibilidad de las paradas en el recorrido de vuelta. En el ejemplo, si un turista quiere ir de la parada 1 a la parada 4, tendría que tomar el bus en la parada 1 y viajar hasta la parada 5, y posteriormente tomar el bus en la parada 5 e ir hasta la parada 4 en el recorrido de vuelta.

Dada la disponibilidad de las paradas en una matriz como la mostrada en el ejemplo, escribir un programa en C que utilice grafos para determinar si un turista puede ir desde una parada p hasta una parada q en el bus turístico. La salida del algoritmo debe ser simplemente si se puede ir desde la parada p a la q.

Para el desarrollo se tendrá en cuenta lo siguiente:

- a) Definición de estructuras en C para el desarrollo del problema (2 puntos)
- b) Construcción del grafo con los datos de entrada. El número de paraderos y la matriz de disponibilidad deben ser ingresados por teclado (3 puntos)

Parte Electiva:

- c) Solución del problema del recorrido. Pueden emplear cualquier algoritmo de recorrido de grafos visto en clase (5 puntos)

Parte Electiva:

Pregunta 3 (5 puntos)

Stock de tienda:

Una tienda de productos almacena la información en un árbol binario de búsqueda en donde la clave es un número entero que es único para cada producto. La empresa tenía un sistema antiguo en donde almacenaba los productos, posteriormente compró un nuevo sistema, y no hicieron la integración de la información. Por lo tanto, la empresa ahora tiene dos colecciones de datos en dos árboles binarios de búsqueda diferentes.

Debido a un proceso de inventario, el gerente de la tienda necesita conocer todos los productos que tiene la empresa ordenados por sus claves. La tienda necesita un ingeniero informático que escriba un programa en C, tal que dados dos árboles binarios de búsqueda, imprima todos los elementos de la colección entera en orden.

Dado que el proceso de inventario tiene que realizarse lo más rápido posible, es necesario que el algoritmo sea de complejidad $O(m + n)$, en donde m y n son los tamaños de los árboles. Además hay una restricción con respecto a la cantidad de espacio extra que se puede usar. Como máximo se puede emplear $O(h_1 + h_2)$ como espacio extra, donde h_1 es la altura del primer árbol y h_2 es la altura del segundo árbol.

NOTA: Para cumplir con las restricciones de complejidad y de espacio, se debe usar una pila para cada árbol.

Profesores del curso: Rony Cueva
 Ivan Sipiran

Pando, 12 de noviembre del 2019