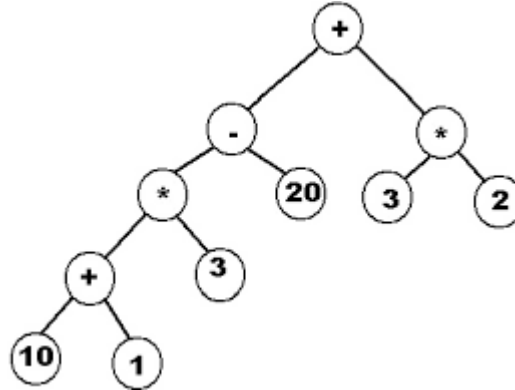


2° RECUPERATORIO DE ALGORITMICA Y ESTRUCTURA DE DATOS II - 2017

1) Dadas dos listas (List) que contienen empleados, elimine de la primer lista todos los empleados que están en la segunda lista. De cada empleado se registra su nombre, edad y legajo. (El legajo es único para cada empleado)

2) Dado el siguiente árbol binario:



- Utilizar el TAD de árbol binario para crear el mismo.
- Recorrer el árbol e insertar todos los elementos en una pila de manera tal que quede en la misma, la notación postfija de la expresión aritmética.
- Implementar un método que retorne una lista de los distintos tipos de operadores que hay en un árbol de expresiones.

3) Agregar al TAD de árbol binario el método **search** que dado un elemento retorna la posición del nodo donde se encuentra el elemento o **null** si no se encuentra.

Position<E> search(E p)

Realizar un ejemplo que utilice los métodos implementados utilizando el árbol del ejercicio anterior.

4) Desarrollar una aplicación que cargue empleados ordenados por legajo en un **Map**. Utilizar la implementación de un árbol binario ordenado. Mostrar con distintos ejemplos el uso del método **subMap()**.

TEORIA (Entregar en una hoja aparte)

- ¿Qué recursos se tienen en cuenta al hacer análisis de algoritmos y cuál es la diferencia con el análisis experimental de algoritmos?
- ¿Qué es un objeto de tipo Iterador y como se implementa el método `iterator()`? Ejemplifique la explicación en Java y haga el correspondiente diagrama sencillo de clases.
- Describe el TAD tabla de búsqueda, cuál es su interfaz y en qué casos es ineficiente.