



UNIDAD TEMÁTICA 4: Árboles Binarios I

TRABAJO DE APLICACIÓN 3

Diseño de algoritmos adicionales para el árbol binario de búsqueda

Se desarrollarán los algoritmos que se indican, prestando atención a completar todos los pasos (lenguaje natural, especificación de pre y post – condiciones, pseudocódigo)

Ejercicio #1

NOTA IMPORTANTE: REALIZAR LAS TAREAS INDICADAS EN EL ORDEN ESTABLECIDO!

EN SUBEQUIPOS:

Sub-Equipo A:

1. Definir **pre y post - condiciones** de un algoritmo para **obtener la altura** de un árbol binario de búsqueda.
2. A partir de las **post-condiciones** definidas, escribir (en lenguaje natural) los casos de prueba pertinentes.
3. Diseñar en pseudocódigo la prueba del algoritmo para **obtener la altura** de un árbol binario de búsqueda.
4. Revisar los casos de prueba escritos en el punto 3)

Sub-Equipo B:

1. Definir pre y post - condiciones de un algoritmo para **obtener el tamaño** de un árbol binario de búsqueda.
2. A partir de las post-condiciones definidas, escribir (en lenguaje natural) los casos de prueba pertinentes
3. Diseñar en pseudocódigo la prueba del algoritmo para **obtener el tamaño** de un árbol binario de búsqueda.
4. Revisar los casos de prueba escritos en el punto 2)

TRABAJO DE APLICACIÓN 3

Ejercicio #2 - test cases para obtenerAltura y obtenerTamaño

PASO 1: TODO EL EQUIPO:

- a) Tener abierto en una computadora el tutorial de JUNIT (<http://www.vogella.com/articles/JUnit/article.html>)
- a) Hacer que el entorno de desarrollo genere los métodos de prueba correspondientes de TElementoAB y TArbolBB para **ObtenerAltura** y **ObtenerTamaño** (sin implementación, con emisión de excepciones)
 - VS Code: <https://code.visualstudio.com/docs/java/java-testing>
- b) Sincronizar las computadoras con el repositorio

PASO 2: EN SUBEQUIPOS:

Sub-Equipo A:

- a) Implementar los “**test cases**” para el método **obtenerTamaño** del árbol.

Sub-Equipo B:

- a) Implementar los “**test cases**” para el método **obtenerAltura** del árbol.

PASO 3:

TODO EL EQUIPO: Integrar en repositorio GIT.

TRABAJO DE APLICACIÓN 3

Ejercicio #3

Agregar funcionalidades al TDA Arbol Binario de Búsqueda.

PASO 1: EN SUBEQUIPOS:

Sub-Equipo A:

- a) Implementar en JAVA el algoritmo para **obtener la altura** del árbol.
- b) Ejecutar los test cases correspondientes, previamente desarrollados
- c) Revisar el código implementado y resolver eventuales errores descubiertos.

Sub-Equipo B:

- b) Implementar en JAVA el algoritmo para **obtener el tamaño** del árbol.
- c) Ejecutar los test cases correspondientes, previamente desarrollados
- d) Revisar el código implementado y resolver eventuales errores descubiertos.

PASO 2: TODO EL EQUIPO

- a) Integrar los cambios en el repositorio
- d) Probar los algoritmos insertando en el árbol varias claves y verificar en papel que ambos algoritmos funcionen correctamente