

## Universidad CENFOTEC



### Escuela de Software

**Código del curso:** BISOFT-11.

**Nombre del curso:** Estructuras de Datos 1.

**Sección:** SCV1.

**Periodo:** C3-2025.

**Docente facilitador:** Romario Salas Cerdas.

### CONSIGNA DEL SEGUNDO ESTUDIO DE CASO

#### 1. Datos generales de la actividad

<b>Tipo de actividad:</b>	Estudio de caso.		
<b>Fecha de entrega:</b>	27 de noviembre del 2025, 11:59pm. <b>Valor porcentual:</b> 15%.		
<b>Formato de entrega:</b>	Archivo comprimido.	<b>Puntaje total:</b>	100.
<b>Individual:</b> Sí.	<b>Grupal:</b> No.		

#### 2. Instrucciones generales

1. Lea cuidadosamente las instrucciones de la actividad. En caso de tener alguna duda, puede consultar con su docente.
2. Esta actividad se desarrolla de manera individual. Cualquier intento de plagio será sancionado de acuerdo con el reglamento académico vigente.
3. Al completar la actividad, el estudiante debe subir un archivo comprimido al buzón del entorno de Moodle designado para ese fin. El archivo comprimido debe contener los archivos de código fuente **.cpp** y **.h** necesarios para la compilación y posterior ejecución del programa solicitado, además de un archivo **.pdf** con el desarrollo de la documentación solicitada.

#### 3. Objetivos o competencias del curso que se evaluarán en la actividad de aprendizaje

<b>Objetivo general o competencia del curso</b>	Construir aplicaciones de software pequeñas y medianas, usando las estructuras de datos lineales, las estructuras de datos jerárquicas y los algoritmos de ordenamiento y búsqueda, para implementar aplicaciones de software más eficientes en tiempo y en recursos de computadora.
---	--

<b>Objetivos específicos que se evalúan</b>	<ul style="list-style-type: none"> <li>• Comprender el funcionamiento y la algorítmica de las estructuras de datos lineales y jerárquicas y de los algoritmos de ordenamiento y búsqueda, mediante el estudio de ejemplos y prácticas en clase, a fin de que el estudiante pueda conocer y distinguir las diferentes estructuras de datos.</li> <li>• Usar las estructuras de datos lineales y jerárquicas, y los algoritmos de ordenamiento y búsqueda, mediante el uso de Tipos Abstractos de Datos y de lenguajes de programación, para resolver un problema con una aplicación de software pequeña o mediana.</li> <li>• Analizar la eficiencia en tiempo y en espacio, de las estructuras de datos y de los algoritmos, usando la notación O, a fin de aplicar este criterio y escoger las estructuras y los algoritmos de menor complejidad.</li> <li>• Diseñar estructuras de datos lineales y jerárquicas por medio de Tipos Abstractos de Datos, para construir aplicaciones de software fáciles de entender, corregir y cambiar.</li> <li>• Elaborar aplicaciones de software eficientes, implementando estructuras de datos lineales y jerárquicas, y algoritmos de búsqueda y ordenamiento, que garanticen el menor uso de recursos y de tiempo.</li> </ul>
---	---

#### 4. Descripción de la actividad

**Contexto:**

**Contexto:**

El segundo estudio de caso consiste en el desarrollo de una investigación en torno a la implementación de los algoritmos de evaluación del factor de balance en los árboles autobalanceados AVL, así como de los algoritmos de rotación que permiten la corrección de los desbalances que aparecen en dichas estructuras.

**Instrucciones:**

El estudiante debe indagar y documentar de forma completa y adecuada el funcionamiento de los algoritmos de evaluación del factor de balance de los nodos de un árbol AVL y de rotación de los nodos para la corrección de desbalances, en al menos tres páginas de texto (Arial 12, interlineado 1.5; no se incluye en dicha extensión la portada ni las

referencias bibliográficas, si las hay), incluyendo representaciones gráficas del paso a paso de dichas operaciones. El estudiante debe investigar también las principales aplicaciones de los árboles AVL, listándolas y explicando por qué los árboles AVL son las estructuras de datos adecuadas para ellas.

Seguidamente, el estudiante debe programar en C++ un árbol AVL que efectúe de forma adecuada la funcionalidad de dicha estructura de datos. Los cuatro métodos para la rotación de los nodos (i.e., **rotarIzquierda()**, **rotarDerecha()**, **rotarIzquierdaDerecha()** y **rotarDerechaIzquierda()**, todos los cuales reciben de parámetro un **NodoAVL\*** y lo retornan tras realizar los ajustes requeridos en la estructura) deben de ubicarse dentro de la clase **NodoAVL**, mientras que la lógica para **buscar()**, **insertar()** y **eliminar()** debe estar contenida en la clase **AVL**. El método **buscar()** funciona exactamente igual que el de un árbol binario de búsqueda ordinario, mientras que los de **insertar()** y **eliminar()** deben ser modificados para que actualicen adecuadamente las alturas de los nodos afectados por la operación realizada. La investigación y búsqueda de implementaciones que realicen estas operaciones de forma adecuada, según la definición dada a la estructura, es parte del trabajo que el estudiante debe llevar a cabo. El estudiante debe entregar las dos parejas de archivos de código fuente (i.e., un archivo de encabezado **.h** y un archivo de operaciones **.cpp** por cada clase mencionada).

El código entregado debe incluir una rutina **main()** que ponga a disposición del usuario un **menu()**, mediante el cual se pueda interactuar de forma eficiente e intuitiva con el árbol AVL desarrollado. El **menu()** debe incluir como mínimo las opciones de buscar, insertar y eliminar un nodo según su llave, además de la visualización de los contenidos del árbol dispuestos en orden.

## 5. Rúbrica

Esta actividad de aprendizaje será evaluada mediante la siguiente rúbrica:

1. Implementación adecuada de la clase **NodoAVL**: **15 puntos**.
2. Implementación adecuada de la clase **AVL**: **35 puntos**.
3. Desarrollo adecuado de la UI requerida: **15 puntos**.
4. Desarrollo adecuado de la documentación solicitada: **35 puntos**.

Total: **100 puntos**.

Criterio	Deficiente (1 punto)	Regular (2 puntos)	Bueno (3 puntos)	Excelente (4 puntos)
Requerimientos y especificaciones	Describe los requerimientos y especificaciones del estudio de caso de forma insuficiente y sin profundidad	Describe los requerimientos y especificaciones del estudio de caso de forma elemental y con poca profundidad	Describe los requerimientos y especificaciones del estudio de caso de forma aceptable y con profundidad	Describe los requerimientos y especificaciones del estudio de caso de forma adecuada y con profundidad
Estructuras de datos	Analiza las estructuras de datos situación planteada de manera inadecuada y con limitaciones	Analiza las estructuras de datos situación planteada de manera básica y con varias limitaciones	Analiza las estructuras de datos situación planteada de manera regular y con alguna limitación	Analiza las estructuras de datos situación planteada de manera adecuada y sin limitaciones
Funcionamiento de estructuras de datos	Propone el funcionamiento de las estructuras de datos de manera apropiada	Propone el funcionamiento de las estructuras de datos de manera elemental	Propone el funcionamiento de las estructuras de datos de manera regular	Propone el funcionamiento de las estructuras de datos de manera apropiada
Estrategia de solución	Desarrolla una estrategia de solución al caso propuesto de manera inapropiada y no lo fundamenta con los tipos abstractos de datos	Desarrolla una estrategia de solución al caso propuesto de manera básica y hay poca fundamentación de los tipos abstractos de datos	Desarrolla una estrategia de solución al caso propuesto de manera aceptable y lo fundamenta con los tipos abstractos de datos	Desarrolla una estrategia de solución al caso propuesto de manera apropiada y lo fundamenta con los tipos abstractos de datos
Informe de la solución	Elabora un informe de la solución del estudio de caso con las pautas y líneas a seguir de manera escasa, con múltiples errores y sin respetar el mínimo de páginas	Elabora un informe de la solución del estudio de caso con las pautas y líneas a seguir de manera elemental, con varios errores y sin respetar el mínimo de páginas	Elabora un informe de la solución del estudio de caso con las pautas y líneas a seguir de manera regular, con algún error y respeta el mínimo de páginas	Elabora un informe de la solución del estudio de caso con las pautas y líneas a seguir de manera adecuada, sin errores y respeta el mínimo de páginas
Entrega de la solución	Realiza la entrega del estudio de caso de manera inadecuada			Realiza la entrega del estudio de caso de manera adecuada