

## Modelo Evaluación de consolidación

Apellido y Nombre: \_\_\_\_\_

Legajo: \_\_\_\_\_ DNI: \_\_\_\_\_

### Ejercicio 1 – Función ContarAparicionesSubliminales

**Consigna:** Implementar la función “ContarAparicionesSubliminales” que cumpla con la post-condición descrita en el código fuente adjunto a la evaluación.

**Criterio de evaluación:** Se evaluará el cumplimiento de los requerimientos descritos en la post-condición de la función (**debe pasar todos los test unitarios dados**). Se evaluará también la descomposición funcional y todo el diseño de código que se realice.

**Material:** main.cpp, TestSuitConsolidacion.h y TestSuitConsolidacion.cpp

**Descripción del material:** Encontrará en el archivo main.cpp un programa que ejecuta una serie de test unitarios sobre la función ContarAparicionesSubliminales.

**Restricción:** No se puede incluir ninguna librería. No se puede modificar nada de arbol.h

---

### Ejercicio 2 – Transacción

**Descripción del Dominio:**

**Transacción y Cuenta:**

Una instancia de Transacción representa una transacción realizada de una cuenta origen a una cuenta destino. Al crearse una transacción esta realiza un débito de una cuenta origen y una acreditación en una cuenta destino por un determinado monto, además esta transacción tiene un identificador único (puede ser numérico o alfanumérico, eso lo puede decidir el/la programador/a).

Lo que le interesa al código cliente de su TDA Transacción es poder saber su identificador, el monto de la transacción, la cuenta origen y la cuenta destino. Además estos datos deben ser inmutables en la transacción.

Una Cuenta tiene asociado un dueño y un saldo disponible. El saldo disponible de una cuenta origen limita que no se puedan hacer transacciones cuando el monto de la transacción es superior al saldo disponible. El saldo es modificado por las transacciones.

**Consigna:**

1. Defina e implemente el TDA Transacción.
2. Defina solamente la interfaz del TDA Cuenta que necesita para realizar la transacción.

### Ejercicio 3

#### Parte A - Árboles

El recorrido en-orden de un árbol binario es

- 8 10 5 100 7 9

El recorrido en pre-orden del mismo árbol es

- 100 10 8 5 9 7

Dibuje el Árbol y responda

1. ¿Es un árbol binario de búsqueda? justifique

2. ¿Cual es la altura? justifique