

Ejercicios Básicos sobre Clases Estudiante y Curso

Descripción

En este conjunto de ejercicios trabajaremos con dos clases en C++: **Estudiante** y **Curso**. La clase **Estudiante** almacena el nombre y la calificación final de un estudiante. La clase **Curso** contiene un arreglo de estudiantes y el nombre del curso.

Ejercicios Propuestos

1. **Mostrar todos los estudiantes**

Implementar un método `mostrarEstudiantes()` en la clase **Curso** que imprima los datos de todos los estudiantes del curso.

2. **Encontrar la calificación más alta**

Crear un método que recorra el arreglo de estudiantes y determine cuál tiene la calificación más alta. Mostrar su nombre y su calificación.

3. **Encontrar la calificación más baja**

Similar al ejercicio anterior, pero para encontrar al estudiante con la calificación más baja.

4. **Calcular el promedio de calificaciones**

Implementar un método que calcule el promedio de todas las calificaciones del curso.

5. **Contar cuántos estudiantes aprobaron y reprobaron**

Crear un método que cuente cuántos estudiantes tienen una calificación mayor o igual a 70 (aprobados) y cuántos tienen menos de 70 (reprobados).

6. **Buscar estudiante por nombre exacto**

Implementar un método que reciba un nombre como parámetro y busque al estudiante exacto. Si lo encuentra, mostrar su calificación; si no, mostrar un mensaje adecuado.

7. **Modificar calificación de un estudiante**

Crear un método que permita modificar la calificación de un estudiante especificando su nombre. Si se encuentra, se actualiza la calificación; si no, se informa que no existe.

8. **Eliminar estudiante (opcional)**

(Para mayor dificultad) Implementar una función que “elimine” a un estudiante poniendo su nombre como “Eliminado” y su calificación en 0.0. No es necesario mover los elementos del arreglo.

9. **Ordenar la lista de estudiantes de menor a mayor o viceversa (opcional)**

Implementar un método que ordene el arreglo de estudiantes según su calificación final. El orden puede ser ascendente (de menor a mayor) o descendente (de mayor a menor), según se requiera. Puedes utilizar el método de ordenamiento burbuja (`bubble sort`) para facilitar la implementación. Al finalizar, mostrar la lista ordenada utilizando el método `mostrarEstudiantes()`.

Sugerencias

- Usa bucles `for` para recorrer los arreglos de estudiantes.
- Utiliza métodos `get` en la clase `Estudiante` si los necesitas (deberás agregarlos si no existen).
- Recuerda inicializar correctamente los valores máximos y mínimos antes de comparar.
- Puedes reutilizar el método `toString()` para mostrar la información de los estudiantes.