

Universidad de Los Andes
Escuela de Ingeniería de Sistemas
Departamento de Computación

PROGRAMACIÓN 2

Clase 9

Junior Altamiranda
altamira@ula.ve

Ejercicio

Realizar la implementación de dos clases:

Instrumento:

Propiedades: nombre, tipo, invertor, anyo, pais

Métodos: Constructor vacío, constructor paramétrico, destructor, lectura, mostrar

Guitarra

Propiedades: Tipo de Guitarra, material, nro de cuerdas.

Métodos: Constructor vacío, constructor paramétrico, destructor, lectura, mostrar.

Guitarra es una subclase de Instrumento.

Escribir un programa principal que permita registrar cierta cantidad de guitarras.

Polimorfismo y Encadenamiento dinámico

- Habilidad de los objetos de distintas clases de responder a un método con el mismo nombre, de acuerdo a un comportamiento específico de la subclase.
- El encadenamiento dinámico (tardío) ocurre cuando, en tiempo de ejecución, se enlaza la referencia al objeto correcto (**v-table**)
- El encadenamiento estático ocurre, por el contrario, en tiempo de compilación
- Para hacer uso del Polimorfismo en C++, se debe utilizar la palabra clave **virtual**, al definir la función que va puede ser sobreescrita en las subclases
- El encadenamiento dinámico tiene su coste: Hace ligeramente más lenta la ejecución del código (10% dicen algunos).

Ejercicio

- 1 Se tienen las clases Persona y Empleado. Empleado es una subclase de Persona.
- Realizar las modificaciones que considere necesarias en las clases y crear en el programa principal el comportamiento polimórfico de un vector a través del encadenamiento dinámico del método showAll() que se encuentra en ambas clases.

Ejercicio

- Se tienen las clases Instrumento y Guitarra respectivamente. Guitarra es una subclase de Instrumento.
- Realizar las modificaciones que considere necesarias en las clases y crear en el programa principal el comportamiento polimórfico de un vector a través del encadenamiento dinámico del método showAll() que se encuentra en ambas clases.

Sobrecarga de operadores

Mecanismo a través del cual, operadores estándar de C++ son usados para otras operaciones de clase.

Los operadores son los que ya existen: No se pueden crear nuevos operadores

Todos los operadores pueden ser sobrecargados, excepto:

., ?:, ::, *

Un ejemplo de un operador “sobrecargado” en C es <<

Ni la precedencia, ni la asociatividad (orden), ni la cantidad de operandos de un operador, pueden ser modificados

Sobrecarga de operadores

Implementación:

```
class algo {  
    public:  
        int operator+(int b);  
};  
  
int algo::operator+(int b) {  
    . . .  
}
```

Ejercicio: El TDA Cadena

- Deseamos definir y construir el TDA Cadena
- Una Clase cadena; que facilite la manipulación de cadena de caracteres en C++, añadiendo métodos y propiedades que nos permitan ahorrar tiempo y facilite nuestro trabajo
- La cadena, la veremos como una secuencia de caracteres

TDA Cadena

(Características a incluir)

Propiedades

- Longitud

- Contenido

Métodos

- Constructores

- Destructor

- Comparar la cadena con otra cadena

- Buscar texto en cadena

- Substraer: Obtener cierta cantidad de caracteres, a partir de cierta posición

- Concatena otra cadena

- Asignación de un nuevo valor