**PROGRAMACIÓN APLICADA A LA AUTOMATIZACIÓN**

**PREPRÁCTICA # 6 Desarrollo de aplicaciones en C++ empleando composición de clases.**

**CAPÍTULO DEL CURSO:** Programación orientada a objetos y estructuras de datos aplicadas a automatización.

**TEMA DE LA ACTIVIDAD:** Creación de programas que usen clases, clases compuestas y objetos.

**NOTA**

**OBJETIVOS DE APRENDIZAJE:**

* Elaborar algoritmos que empleen composición de clases para la creación de programas en C++ que resuelvan problemas prácticos.

**DURACIÓN:** 120 minutos

**MATERIALES Y HERRAMIENTAS:**

* Computadora con Qt Creator correctamente instalado o algún otro editor de C o de texto.

**INTRODUCCIÓN**

El lenguaje C++ debe su popularidad al uso de la Programación Orientada a Objetos. La POO, utilizando el concepto de encapsulamiento, es un paradigma de programación que busca acercar al programador al problema y su solución, y alejarlo de los detalles de la computadora y el lenguaje. Además, tiene la gran ventaja de la reutilización del código previamente diseñado lo que hace más eficiente el trabajo. Dentro de la POO, por ahora, se pueden destacar elementos como las “Clases”, que incluyen datos y también funciones miembros, constructores y destructor. Una manera de reutilización del código previamente diseñado en POO es la composición de clases, que permite construir nuevas clases utilizando otras como sus partes. En esta parte, se realizará un ejercicio bastante completo de uso de clases y su composición. Se debe emplear como bibliografía la conferencia VI o sesión de clase 6 y la propia bibliografía allí propuesta.

**DESCRIPCIÓN DE LA PRÁCTICA:** El estudiante debe implementar la solución del ejercicio, agregarla al informe y subirlo antes de la fecha y hora prevista.

**PROCEDIMIENTO:**

1. Realice el programa en lenguaje C++ necesario para resolver la pregunta o problema que se encuentra posteriormente en este documento.
2. Incorpore dicho programa al informe de esta prepráctica en formato texto.
3. Suba el informe a AulaVirtual antes del comienzo de la práctica.

**Ejercicio 1**

Defina una clase **Computer** que está formada por los siguientes miembros:

1. Una numero entero que represente el ancho de resolución de la pantalla
2. Un numero entero que represente el alto de la resolución de la pantalla
3. Un número entero que represente el tamaño de su disco duro.
4. Un número entero que represente el tamaño de memoria.
5. Un número real que represente la velocidad del procesador.
6. Defina un constructor por omisión para esta clase donde: la resolución sea 1024X768, la memoria ram sea de 2GBytes, el disco duro de 500 GBytes, la velocidad 1,77 GHz.

Implemente los métodos u operaciones de la clase que permitan:

* Modificar todos los atributos del objeto declarado como Computer
* Devolver los atributos de manera individual
* Que permita imprimir la información completa del objeto Computer

1. Validar en el constructor los atributos ancho y alto para las resoluciones de: **340 X 600, 800 X 600 y 1024 X 768**, **1920X1080** (Utilice encapsulamiento)

En el programa principal, cree una variable de tipo arreglo dinámico de punteros a computadoras (no es un arreglo dinámico de computadoras sino de punteros a computadoras, debe usar **\*\***). Cree una computadora dinámicamente y asígnela al primer puntero del arreglo. Imprima en pantalla la información apuntada por este primer elemento del arreglo.

**IMPLEMENTACIONES Y RESULTADOS**

**Ejercicio #1:**

[Texto del código]

**CONCLUSIONES Y RECOMENDACIONES**

**BIBLIOGRAFIA**

[1]. Deitel and Deitel. Como programar en C/C++. Segunda edición o superior.

[2]. García de Jalón, J.; y otros. Aprenda C++ como si estuviera en primero. Universidad de Navarra.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| **INICIAL** | **EN DESARROLLO** | **DESARROLLADO** | **EXCELENTE** |

**Firma del Profesor**