



Lenguaje C++

Objetivos

- Aprender el paradigma de Programación Orientado a Objetos.
- Comprender y aplicar las distintas características de la programación orientada a objetos.
- Conocer el entorno de trabajo e incorporar las herramientas básicas para la programación orientada a objetos.
- Desarrollar aplicaciones en el paradigma de orientación a objetos utilizando el lenguaje C++.

Prerrequisitos

Para realizar este curso es necesario tener conocimientos básicos de programación o haber realizado el curso Diagramación Lógica y lenguaje C.

Modalidad

Curso Distancia Sincrónico

Duración

16 semanas

CONTENIDOS DEL CURSO

Unidad I: Introducción a la programación orientada a objetos

Objetivos específicos:

Comprender la estructura básica de la programación en C++.

Contenido:

Concepto de Programación Orientada a Objetos.
Características: abstracción, encapsulamiento, herencia y polimorfismo.
Concepto de clase, objeto, mensaje, método.
Programación Lineal o Estructurada Vs Programación Orientada a Objetos.
Entorno de desarrollo. Diferencias entre C y C++.
Estructura de un programa.

Unidad II: Clases y abstracción de datos. Parte 1

Objetivos específicos:

Entender los conceptos relacionados con el encapsulamiento y ocultamiento de datos. Entender conceptualmente la abstracción de datos y los tipos de datos abstractos. Entender cómo se crean, utilizan y destruyen los objetos de una clase. Controlar el acceso a los datos miembros y a las funciones miembros de los objetos. Comenzar a valorar la orientación a objetos.



Contenido:

Definir una estructura. Acceso a miembros de una estructura Alcance de una clase y acceso a los miembros de la clase.
Separación de la interfaz y la implementación.
Control de acceso a los miembros. Funciones de acceso y funciones de utilidad.
Inicialización de los objetos de una clase: constructores Uso de argumentos predeterminados con constructores. destructores. Uso de funciones set y get.

Unidad III: Clases y abstracción de datos. Parte 2

Objetivos específicos:

Crear y destruir objetos dinámicamente. Especificar objetos const y funciones miembros const.
Comprender el propósito de las funciones friend y de las clases friend. Aprender a utilizar los datos miembros y las funciones miembros static. Conocer el manejo del apuntador this.

Contenido:

Objetos y funciones constantes.
Composición: objetos como miembros de clases.
Funciones friend.
Uso del apuntador this.
Asignación dinámica de memoria: operadores new y delete.
Miembros de clase static.

Unidad IV: Sobrecarga de operadores

Objetivos específicos:

Comprender cómo redefinir (sobrecargar) los operadores para que trabajen como nuevos tipos de datos.

Contenido:

Fundamentos de la sobrecarga de operadores.
Funciones de operador como miembros de la clase o como funciones friend.
Sobrecarga de los operadores de inserción y de extracción de flujo.
Sobrecarga de operadores unarios.
Sobrecarga de operadores binarios.

Unidad V: Herencia

Objetivos específicos:

Crear nuevas clases heredando las clases ya existente. Entender la manera en que la herencia promueve la reutilización del software. Comprender los conceptos de clase base y clase derivada. Aprender a utilizar la herencia múltiple para derivar una clase a partir de varias clases bases.

Contenido:

Introducción. Concepto.
Clases base y clases derivadas.
Miembros protected.
Constructores y destructores en clases derivadas.
Tipos de herencia: pública, privada y protegida.



Unidad VI: Polimorfismo

Objetivos específicos:

Comprender los conceptos de polimorfismo. Aprender a declarar y utilizar funciones virtuales para facilitar el polimorfismo. Comprender la distinción entre clases abstractas y clases concretas. Aprender a declarar funciones virtuales puras para crear clases abstractas.

Contenido:

Relaciones entre los objetos en una jerarquía de herencia: invocación de funciones de clase base a partir de objetos de clases derivadas; llamadas a funciones miembro de clases derivadas mediante apuntadores de clase base. Funciones virtuales.

Clases abstractas

Bibliografía:

- ✦ Francisco. Javier Ceballos Sierra - (2007) – *Programación orientada a objetos con C++ 4ª Edición.* - Madrid - Editorial. Ra-ma.
- ✦ H. M. Deitel y P. J. Deitel - (2003) - *Cómo programar en C++ 4º Edición* - México - Editorial Prentice Hall.