



# PROGRAMACIÓN y Aplicaciones para móviles



## Programación en C++

### CARACTERÍSTICAS

#### DURACIÓN

64 horas

#### OBJETIVOS

Los lenguajes C y C++ forman la base de programación de todas las carreras, constituyendo los lenguajes de programación para desarrollar múltiples aplicaciones a todos los niveles. Al finalizar el curso, el alumno dominará en profundidad los lenguajes C y C++.

#### A QUIÉN VA DIRIGIDO

Todas aquellas personas, interesadas en adquirir una sólida base para el desarrollo de aplicaciones en lenguaje C o C++ de cualquier área profesional que demande aplicaciones a alto nivel o a nivel de máquina.

#### MODALIDADES

Presencial

#### REQUISITOS

El plan de estudios asume que los estudiantes tengan conocimientos previos de informática a nivel de usuario medio. Se espera de ellos una buena capacidad de lectura y expresión escrita, así como un deseo de aprender el programa de estudios.:

- Imprescindibles:
  - + Disponer de horas adicionales (entre 5 y 10) a la semana para realizar ejercicios
  - + Disponer de un manejo fluido de internet
- Recomendables:
  - + Ser capaz de leer textos en inglés y comprender la idea principal del mismo

#### PROGRAMA

##### Programación en C

- Primeras órdenes en lenguaje C
- Variables y constantes
- Estructura básica de un programa
- Palabras clave en C. Identificadores y variables
- Nombres de identificadores
- Tipos de datos. Modificadores de tipo
- Declaración de variables. Constantes. Inicialización de variables
- Operadores. Expresiones. Cambio de tipo en las expresiones
- Sentencias de Control. If, For, Switch, while, do while, Break, Continue
- Funciones. Características generales de las funciones
- Paso por valor y paso por dirección
- Argumentos de la función main
- Punteros. Cómo funcionan los punteros. Operaciones con punteros





- Paso de punteros como parámetros de función. Punteros y vectores
- Arrays. Matrices de una dimensión (vectores)
- Arrays bidimensionales (matrices). Arrays multidimensionales (poliedros)
- Paso de Arrays como parámetros de función
- Cadenas. Cadenas como vectores de caracteres
- Operaciones con cadenas (comparación, copia e concatenación)
- Conversión entre datos numéricos y cadenas. Arrays de cadenas
- El procesador. #INCLUDE, #DEFINE. Macros predefinidas
- Compilación condicional. #UNDEF, #ERROR, #LINE, #PRAGMA
- Estructuras y uniones. Definición de estructuras. Cómo utilizar una estructura
- Creación dinámica de estructuras
- Estructuras de datos: nodos y listas enlazadas
- Ficheros. Operaciones con ficheros
- Declaración de punteros a ficheros
- Abrir un fichero. Cerrar un fichero. Mantenimiento de un fichero
- Asignación dinámica de memoria. Listas enlazadas o pilas. Crear lista
- Mostrar lista completa. Insertar elementos. Buscar elementos. Borrar elementos
- Librería de funciones de C
- Funciones de cadena, matemáticas, de fecha y hora, de pantalla en modo textos
- Funciones de conversión de datos, de disco y directorio

## Programación en C++

- El primer programa C++
- Modificaciones menores con respecto a C
- Visibilidad de variables. Conversiones explícitas de tipo
- Especificador inline para función. Variables de tipo referencia
- Operadores new y delete. Nueva forma de realizar la entrada/salida
- Clases. Declaración de clases. Constructores y destructores
- Clases y funciones friend. El puntero this. Variables de clase
- Herencia. Definición e necesidad de Herencia
- Constructores de clases derivadas. Herencia simple y Herencia múltiple
- Clases base virtuales. Polimorfismo. Sobrecarga de operadores
- Polimorfismo en las clases. Funciones virtuales
- Entrada/salida en C++. Entrada/salida con formato
- Sobrecarga de los operadores de entrada/salida
- Entrada/salida de ficheros. Aspectos avanzados. Plantillas
- Plantillas de función. Plantillas en clases
- Miembros y constructores plantilla
- Clases plantilla. Herencia con plantillas
- Introducción a Standard Template Library (STL) de C++
- Contenedores. Iteradores. Algoritmos
- Construyendo y desplegando Ensamblados
- Crear un ensamblado con el Compilador en C++
- Utilizar ildasm. Desplegar un ensamblado localmente
- Crear y desplegar un ensamblado compartido. Código legado
- Escribir código que interactúa con Win32
- Exponer componentes COM legados a clientes C++
- Exponer componentes C++ a clientes COM legados