

### Ficha Técnica

<b>Titulación:</b>	Grado en Ingeniería Informática		
<b>Plan BOE:</b>	BOE número 108 de 6 de mayo de 2015		
<b>Asignatura:</b>	Fundamentos de Programación		
<b>Módulo:</b>	Programación y Estructuras de Datos		
<b>Curso:</b>	1º	<b>Créditos ECTS:</b>	6
<b>Tipo de asignatura:</b>	Básica	<b>Tipo de formación:</b>	Teórica y práctica

### Presentación

En la asignatura de formación básica denominada “Fundamentos de Programación” se dan las bases necesarias para conocer las técnicas de programación modernas a través de lenguajes de alto nivel estructurados. Además, se explicarán los diferentes enfoques y herramientas para afrontar la realización de programas informáticos.

En la primera parte de la asignatura se exponen los conceptos de manera teórica. A través de pseudocódigo, se explicarán las estructuras básicas y reglas fundamentales para la realización de programas. En la segunda parte de la asignatura se realizará la aplicación de los conceptos adquiridos en la primera parte mediante la utilización de un lenguaje de programación concreto. El lenguaje elegido por su versatilidad y utilidad es C++.

Como resultado del estudio de la asignatura se espera que el estudiante sea capaz de realizar la implementación y diseño de programas informáticos independientemente de la plataforma que se vaya a utilizar de manera clara y sencilla.

### Competencias y/o resultados del aprendizaje

- Realizar programas informáticos.
- Definir algoritmos.
- Saber implementar las distintas estructuras de datos y sus operaciones.
- Determinar la eficiencia de un algoritmo.

### Contenidos Didácticos

- 1 Unidad didáctica 1. Concepto de programa informático.
  - 1.1 Introducción a las computadoras
  - 1.2 La programación
  - 1.3 Concepto de programa
  - 1.4 El pseudocódigo
    - 1.4.1 Reglas generales del pseudocódigo
  - 1.5 Estructuras de control
  - 1.6 Secuencia
  - 1.7 Condiciones
    - 1.7.1 Estructura condicional simple
    - 1.7.2 Estructura condicional doble
    - 1.7.3 Estructura condicional múltiple
  - 1.8 Estructuras de control: repeticiones

- 1.8.1 Estructura de tipo Mientras
- 1.8.2 Estructura de tipo Hasta
- 1.8.3 Estructura de tipo Para
- 2 Unidad didáctica 2. Algoritmos y sistemas de representación de un programa.
  - 2.1 Algoritmo
    - 2.1.1 Características básicas de un algoritmo
    - 2.1.2 Algoritmo de Euclides
    - 2.1.3 Algoritmos computables y no computables
  - 2.2 Diagramas de flujo y ordinogramas
    - 2.2.1 El diagrama de flujo
    - 2.2.2 El ordinograma
- 3 Unidad didáctica 3. Tratamiento informático de un problema.
  - 3.1 Introducción
  - 3.2 Definición de los requisitos del problema
  - 3.3 Análisis
  - 3.4 Diseño
    - 3.4.1 Diseño general
    - 3.4.2 Diseño detallado
  - 3.5 Codificación
  - 3.6 Pruebas
  - 3.7 Mantenimiento
- 4 Unidad didáctica 4. La programación y los lenguajes informáticos.
  - 4.1 La programación
  - 4.2 Paradigmas de programación
    - 4.2.1 Tipos de paradigmas de programación
  - 4.3 Paradigma imperativo
    - 4.3.1 Lenguajes imperativos orientados a expresiones y a sentencias
  - 4.4 Los lenguajes de programación
    - 4.4.1 Lenguajes de máquina
    - 4.4.2 Lenguajes de bajo nivel
    - 4.4.3 Lenguajes de alto nivel
    - 4.4.4 Historia de los lenguajes de programación
- 5 Unidad didáctica 5. El lenguaje imperativo C++.
  - 5.1 Introducción
  - 5.2 Historia de C++
  - 5.3 Características generales de C++
  - 5.4 Elementos básicos de C++
  - 5.5 Estructura de un programa C++
  - 5.6 Palabras reservadas de C++
  - 5.7 Comentarios en C++
- 6 Unidad didáctica 6. Expresiones.
  - 6.1 Introducción
  - 6.2 Constantes
    - 6.2.1 Constantes enteras
    - 6.2.2 Constantes reales
    - 6.2.3 Constantes lógicas
    - 6.2.4 Constantes carácter
    - 6.2.5 Constantes cadenas de caracteres
  - 6.3 Variables
    - 6.3.1 Nombres de variables
    - 6.3.2 Declaraciones
    - 6.3.3 Inicializaciones

- 6.3.4 Identificadores constantes
- 6.4 Tipos básicos de datos
  - 6.4.1 Tipo entero
  - 6.4.2 Modificadores de tipo
  - 6.4.3 Tipo real
  - 6.4.4 Tipo carácter
  - 6.4.5 Tipo lógico
  - 6.4.6 Tipo vacío
  - 6.4.7 Punteros
  - 6.4.8 Referencias
  - 6.4.9 Declaraciones en bloques
  - 6.4.10 Conversiones de tipos
- 6.5 Operadores
  - 6.5.1 Operadores y expresiones
  - 6.5.2 Precedencia y asociatividad de operadores
  - 6.5.3 Operadores y expresiones aritméticas
  - 6.5.4 Operadores y expresiones relacionales y lógicas
  - 6.5.5 Operadores y expresiones de asignación
  - 6.5.6 Operadores y expresiones sobre bits
  - 6.5.7 Operadores y expresiones para el manejo de punteros
  - 6.5.8 Otros operadores
- 7 Unidad didáctica 7. Sentencias y entrada/salida básica.
  - 7.1 Introducción
  - 7.2 Sentencias básicas
    - 7.2.1 Asignación
    - 7.2.2 Llamada a funciones
    - 7.2.3 Sentencia return
    - 7.2.4 Sentencia vacía
  - 7.3 Sentencias de control de flujo
    - 7.3.1 Estructura secuencial
    - 7.3.2 Sentencias condicionales
    - 7.3.3 Sentencias repetitivas
    - 7.3.4 Sentencias de salto
  - 7.4 Entrada/salida básica
- 8 Unidad didáctica 8. Funciones.
  - 8.1 Introducción
  - 8.2 Funciones en C++
    - 8.2.1 Declaración de funciones
    - 8.2.2 Definición de funciones
    - 8.2.3 Funciones de librería
    - 8.2.4 Funciones en línea
  - 8.3 Parámetros
    - 8.3.1 Parámetros formales
    - 8.3.2 Parámetros actuales
    - 8.3.3 Paso de parámetros por valor
    - 8.3.4 Paso de parámetros por referencia
    - 8.3.5 Paso por dirección
    - 8.3.6 Parámetros por omisión
  - 8.4 Alcance y visibilidad
    - 8.4.1 Definición de alcance
    - 8.4.2 Variables locales
    - 8.4.3 Variables globales

- 8.4.4 Variables estáticas
- 8.4.5 Resumen del modo de almacenamiento de variables
- 8.5 Recursividad
- 8.6 Sobrecarga de funciones
- 9 Unidad didáctica 9. Tipos avanzados de datos.
  - 9.1 Introducción
  - 9.2 Vectores y matrices
    - 9.2.1 Vectores
    - 9.2.2 Matrices
    - 9.2.3 Representación interna de vectores y matrices
  - 9.3 Punteros
    - 9.3.1 Declaración de punteros
    - 9.3.2 Operador puntero y dirección
    - 9.3.3 Asignación y liberación de memoria dinámica
    - 9.3.4 Aritmética de punteros
    - 9.3.5 Punteros y vectores
  - 9.4 Cadenas
    - 9.4.1 Declaración de cadenas de caracteres
    - 9.4.2 Inicialización
    - 9.4.3 Uso de cadenas de caracteres
  - 9.5 Estructuras y uniones
    - 9.5.1 Concepto de estructura
    - 9.5.2 Declaración de estructuras
    - 9.5.3 Inicialización de estructuras
    - 9.5.4 Uso de estructuras
    - 9.5.5 Campos de bits
    - 9.5.6 Concepto de unión
    - 9.5.7 Inicialización de uniones
    - 9.5.8 Uso de uniones
  - 9.6 Definiciones de tipos
  - 9.7 Tipos enumerados
    - 9.7.1 Concepto de tipo enumerado
    - 9.7.2 Declaración de tipos
    - 9.7.3 Uso de tipos enumerados
- 10 Unidad didáctica 10. El preprocesador de C++. Librerías.
  - 10.1 Introducción
  - 10.2 Preprocesador de C++
    - 10.2.1 Nombres simbólicos y macros
    - 10.2.2 Inclusión de ficheros
    - 10.2.3 Compilación condicional
    - 10.2.4 Otras directivas
  - 10.3 Librerías
    - 10.3.1 Cadenas de caracteres
    - 10.3.2 Librerías de entrada/salida
    - 10.3.3 Librería cstdlib
    - 10.3.4 Librería cmath
    - 10.3.5 Librería ctype
    - 10.3.6 Errores
    - 10.3.7 Librerías limits.h y float.h
    - 10.3.8 Librería stl

## Contenidos Prácticos

Durante el desarrollo de la asignatura se realizarán las siguientes actividades prácticas:

- Prácticas de programación con pseudocódigo.
- Prácticas de programación iniciales con C++.
- Prácticas de programación con expresiones, entrada/salida y funciones.
- Prácticas de programación con tipos avanzados de datos mediante la inclusión de distintos ficheros fuente.

## Evaluación

El sistema de evaluación del aprendizaje de la UDIMA contempla la realización de diferentes tipos de actividades de evaluación y aprendizaje. El criterio de valoración establecido se detalla a continuación:

Actividades de aprendizaje	10%
Controles	10%
Actividades de Evaluación Continua (AEC)	30%
Examen final presencial	50%
<b>TOTAL</b>	<b>100%</b>

## Bibliografía

- Alonso Amo, F. , Fuertes Castro, J.L., Martínez Normand, L. (2009). *Fundamentos de programación*. Madrid: Ed. Udimia.
- Dijkstra, E.W. (1975). *A Discipline Of Programming*. Ed. Prentice-Hall,
- Knuth, Donald. E. (1980). *Algoritmos fundamentales*. Ed. Reverté.
- Alonso, F.; Segovia, J.; Martínez, L. (2005). *Introducción a la ingeniería del software: Modelos de desarrollo de programas*. Delta Publicaciones.
- Alonso, F. Y Segovia, F.J. (1995). *Entornos y metodologías de programación*. Madrid: Ed. Paraninfo.
- Pierce, B. (2002). *Types and programming languages*. MIT Press.
- Watt, D.A. (1990). *Programming language, concepts and paradigms*. London: Prentice Hall Int.
- Ellis, M.A. Y Stroustrup, B. (1991). *The annotated C++ reference manual*. Ed. Addison-Wesley.
- Lippman, S.B. Y Lajoie, J. (2005). *C++ primer*. Ed. Addison-Wesley.
- Alonso, F.; Antonio, A.; Dieste, Ó.; Fuentes, P.; Fuertes, J.L.; López, G.; González,
- Á.L.; Martínez, L.; Medina, T.; Montes, C. Y Pazos, J. (1999). *Programación sin secretos*. Madrid: Cultural.