

## GUIADIDÁCTICA DEL MÓDULO PROFESIONAL

### NOBRE DEL CICLO

Desarrollo de Aplicaciones Multiplataforma /  
Desarrollo de Aplicaciones Web

### CÓDIGO Y NOMBRE DEL MÓDULO PROFESIONAL

MP03A Programación A

### MODALIDAD

NO PRESENCIAL

### DURACIÓN

165 horas (165 horas + 0 hld)



## ÍNDICE

<b>1 INTRODUCCIÓN .....</b>	<b>3</b>
<b>2 RESULTADOS DE APRENDIZAJE, CRITERIOS DE EVALUACIÓN Y CONTENIDOS .....</b>	<b>3</b>
<b>3 EVALUACIÓN .....</b>	<b>6</b>
<b>4 TEMPORALIZACIÓN .....</b>	<b>9</b>

# 1INTRODUCCIÓN

El objetivo general de este módulo consiste en aprender los fundamentos de la programación y las diferentes estructuras de control que están al alcance de los programadores tales como variables, bucles, condicionales o arrays. También nos centraremos en la estructuración y reutilización del código a través de la creación y el uso de métodos, y se tratará la persistencia a través del trabajo con ficheros.

## 2 RESULTADOS DE APRENDIZAJE, CRITERIOS DE EVALUACIÓN Y CONTENIDOS

### UF 1 Programación estructurada(85 horas)

#### Resultados de aprendizaje y criterios de evaluación

**RA1.** Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación usado

#### Criterios evaluación

- a) Identifica los bloques que componen la estructura de un programa informático.
  - b) Crea proyectos de desarrollo de aplicaciones y usa entornos integrados de desa- rrollo.
  - c) Identifica los distintos tipos de variables y la utilidad específica de cada uno de ellos.
  - d) Modifica el código de un programa para crear y utilizar variables.
  - e) Crea y utiliza constantes y literales.
  - f) Clasifica, reconoce y utiliza en expresiones los operadores del lenguaje.
  - g) Comprueba el funcionamiento de las conversiones de tipos explícitos y implíci- tos.
- Introduce comentarios en el código.

#### Contenidos

- 1. **Estructura de un programa informático.**
  - 1.1 Bloques de un programa informático.
  - 1.2 Proyectos de desarrollo de aplicaciones. Entornos integrados de desarrollo.

- 1.3 Variables. Tipos y utilidad.
- 1.4 Utilización de variables.
- 1.5 Constantes. Tipos y utilización.
- 1.6 Operadores del lenguaje de programación.
- 1.7 Conversiones de tipos de datos.
- 1.8 Comentarios en el código.

**RA2.** Utiliza correctamente tipos de datos simples y compuestos usando las estructuras de control adecuadas

### **Criterios evaluación**

- a) Describe los fundamentos de la programación.
- b) Escribe algoritmos simples.
- c) Analiza y diseña los posibles algoritmos para la resolución de problemas.
- d) Escribe y prueba programas sencillos reconociendo y aplicando los fundamentos de la programación.
- e) Utiliza estructuras de datos simples y compuestos.
- f) Escribe y prueba código que use las estructuras de selección.
- g) Utiliza correctamente las distintas estructuras de repetición disponibles.
- h) Reconoce las posibilidades de las sentencias de salto.
- i) Realiza operaciones básicas, compuestas y de tratamiento de caracteres.
- j) Revisa y corrige los errores aparecidos en los programas.
- k) Comenta y documenta adecuadamente los programas realizados.
- l) Utiliza un entorno integrado de desarrollo en la creación y compilación de programas simples..

### **Contenidos**

- 2. Tipos de datos simples y compuestos. Programación estructurada**
- 2.1 Fundamentos de programación.
- 2.2 Introducción a la algorítmica.
- 2.3 Diseño de algoritmos.
- 2.4 Prueba de programas.
- 2.5 Tipos de datos simples y compuestos.
- 2.6 Estructuras de selección.
- 2.7 Estructuras de repetición.
- 2.8 Estructuras de salto.
- 2.9 Tratamiento de cadenas.
- 2.10 Depuración de errores.
- 2.11 Documentación de los programas.
- 2.12 Entornos de desarrollo de programas.

## UF 2. Diseño Modular (50 horas)

### Resultados de aprendizaje y criterios de evaluación

**RA1.** Escribe y prueba programas sencillos reconociendo y aplicando los fundamentos de la programación modular

#### Criterios evaluación

- a) Analiza los conceptos relacionados con la programación modular.
- b) Analiza las ventajas y la necesidad de la programación modular.
- c) Aplica el concepto de análisis descendiente en la elaboración de programas.
- d) Modulariza correctamente los programas realizados.
- e) Realiza correctamente las llamadas a funciones y su parametrización.
- f) Tiene en cuenta el alcance de las variables en las llamadas a las funciones.
- g) Prueba, depura, comenta y documenta los programas.
- h) Define el concepto de librería y su utilidad.
- i) Utiliza librerías en la elaboración de programas.
- j) Conoce las nociones básicas de la recursividad i sus aplicaciones clásicas

#### Contenidos

##### 1. Programación modular

- 1.1 Concepto
- 1.2 Ventajas y inconvenientes.
- 1.3 Análisis descendiente (Top down).
- 1.4 Modularización de programas.
- 1.5 Llamadas a funciones. Tipos y funcionamiento.
- 1.6 Ámbito de las llamadas a funciones.
- 1.7 Prueba, depuración y comentarios de programas.
- 1.8 Concepto de librerías.
- 1.9 Utilización de librerías.
- 1.10 Introducción a la recursividad.

## UF 3. Fundamentos de gestión de ficheros (30 horas)

### Resultados de aprendizaje y criterios de evaluación

**RA1.** Diseña, prueba y documenta programas que realizan distintas operaciones sobre ficheros, documentando el programa y las pruebas

#### Criterios evaluación

- a) Reconoce los conceptos relacionados con ficheros
- b) Reconoce los distintos tipos de ficheros.
- c) Establece y diferencia las operaciones a realizar sobre los ficheros en el lenguaje de programación usado.
- d) Utiliza correctamente diferentes operaciones sobre ficheros.
- e) Modulariza adecuadamente los programas que gestionan ficheros.
- f) Diseña, prueba y documenta programas simples que gestionan ficheros.

#### Contenidos

##### 1. Gestión de Ficheros

- 1.1 Concepto y tipos de ficheros.
- 1.2 Operaciones sobre ficheros secuenciales y relativos.
- 1.3 Diseño de programas de gestión de ficheros.
- 1.4 Modularización de las operaciones sobre ficheros

## 3 EVALUACIÓN

Todos los módulos formativos del ciclo están compuestos por Unidades Formativas (UFs) que serán evaluadas de forma independiente. Así, para poder aprobar el módulo, se deberán superar todas y cada una de las UFs que lo componen.

La media ponderada de todas las calificaciones obtenidas en las diferentes unidades formativas dará como resultado la nota final del módulo:

$$\text{Nota Final}_{MP} = 0,28 \cdot \text{NOTA}_{UF1} + 0,17 \cdot \text{NOTA}_{UF2} + 0,10 \cdot \text{NOTA}_{UF3} + 0,12 \cdot \text{NOTA}_{UF4} + 0,23 \cdot \text{NOTA}_{UF5} + 0,10 \cdot \text{NOTA}_{UF6}$$

La nota de cada Unidad Formativa se obtendrá a través de la evaluación continua, que tendrá lugar en modalidad online a lo largo de todo el semestre, y de la prueba escrita presencial, que se realizará al final del semestre.

La evaluación continua es la base para la evaluación académica del alumnado. Así, existe la obligación de seguir un determinado ritmo de estudio y a realizar, y entregar, las actividades y prácticas (PACs) propuestas en las fechas establecidas.

En relación a las PACs, hay algunos aspectos que deben tenerse en cuenta:

- Las PACs se deben entregar dentro de los plazos establecidos.
- Si el alumno no entrega las PACs en los plazos establecidos, la puntuación máxima que podrá obtener será de 5.
- Una vez publicada la solución de la práctica, no se podrá realizar ninguna entrega.
- Si no se entrega una PAC, la calificación equivaldrá a un 0.
- Para superar la evaluación no sólo se deben presentar las PACs, sino que éstas deben adecuarse a los objetivos planteados por el profesorado y obtener una calificación positiva.

La corrección de las PACs la llevará a cabo el propio docente, que publicará la solución de las prácticas 2 días después de la fecha de entrega. El resultado obtenido se comunicará, como máximo, 7 días más tarde de la publicación de la solución.

Seguir la evaluación continua repercutirá positivamente en la calificación final de la Unidad Formativa. Si se obtiene una buena nota, podrá mejorarse en la prueba escrita final (PEF), ponderando de la siguiente manera:

$$\text{NOTA FINA}_{\text{UF}} = 40\% \text{ NOTA FINAL PACs} + 60\% \text{ PEF}$$

La evaluación continua únicamente se tendrá en cuenta cuando el alumno/a se presente por primera vez a la prueba escrita final, que tendrá lugar al final de cada semestre y que será presencial y obligatoria para todo el alumnado.

En caso de que un alumno/a no supere, o no se presente, a la prueba de evaluación final, podrá volver a hacerlo en la prueba escrita del final del próximo semestre.

Las pruebas escritas finales se estructurarán según las diferentes Unidades Formativas y consistirán en diversos ejercicios: preguntas tipo

test, otras más abiertas, desarrollo de algún pequeño proyecto a partir de un supuesto...

La calificación final de las UFs, así como la del módulo, es numérica del 1 al 10.

**Calificación módulo profesional:**

$$\text{Nota Final}_{MP} = 0,28 \cdot \text{NOTA}_{UF1} + 0,17 \cdot \text{NOTA}_{UF2} + 0,10 \cdot \text{NOTA}_{UF3} + 0,12 \cdot \text{NOTA}_{UF4} + 0,23 \cdot \text{NOTA}_{UF5} + 0,10 \cdot \text{NOTA}_{UF6}$$

Para poder calcular la nota final del módulo, todas las UFs deben estar aprobadas con una calificación igual o superior a 5.

**Calificación UFs:**

$$\text{NOTA}_{FINAL_{UF}} = 60\% \text{ PEF} + 40\% \text{ PACs}$$

Será imprescindible obtener una calificación mayor a 5 en la PEF para poder superar la UF.



## 4 TEMPORALIZACIÓN

Las Unidades Formativas se realizarán según el orden especificado a continuación:

UF1	
	UF2
UF3	

MP03 A. Programación Básica (165 h)			
Unidades Formativas	Duración	Fecha Inicio	Fecha finalización
UF1. Programación Estructurada	85 horas	03/10/2016	22/12/2016
UF2. Diseño modular	50 horas	14/11/2016	07/01/2017
UF3. Fundamentos de gestión de ficheros	30 horas	30/12/2016	20/01/2017

Prueba escrita final	Fechas previstas
PEF enero (ordinaria)	Semana del 30 de enero (horario y sede a concretar)
PEF extraordinaria	Septiembre (Única sede Lleida)
	Final 1S 2016-2017 (Todas las sedes)