



## Grado en Ingeniería Informática

### DATOS DE LA ASIGNATURA

**Nombre:**

Metodología de la Programación

**Denominación en inglés:**

Programming Metodology

**Código:**

606010201

**Carácter:**

Obligatorio

**Horas:**

**Totales**

**Presenciales**

**No presenciales**

**Trabajo estimado:**

150

60

90

**Créditos:****Grupos reducidos**

**Grupos grandes**

**Aula estándar**

**Laboratorio**

**Prácticas de campo**

**Aula de informática**

3

0

0

0

3

**Departamentos:**

Tecnologías de la Información

**Áreas de Conocimiento:**

Ciencias de la Computación e Inteligencia Artificial

**Curso:**

2º - Segundo

**Cuatrimestre:**

Primer cuatrimestre

### DATOS DE LOS PROFESORES

**Nombre:****E-Mail:****Teléfono:****Despacho:**

\*Márquez Hernández,  
Antonio Ángel

amarquez@dti.uhu.es

959217388

ETSI 124

Rodríguez Roman, Miguel  
Ángel

miguel.rodriguez@dti.uhu.e  
s

959217372

134 / Escuela Técnica  
Superior de Ingeniería / El  
Carmen

Paredes Parejo, Francisco Luis	francisco.paredes@dti.uhu. es		144/Escuela Técnica Superior de Ingeniería/El Carmen
-----------------------------------	----------------------------------	--	--

\*Profesor coordinador de la asignatura

[Consultar los horarios de la asignatura](#)

## 1. Descripción de contenidos

### 1.1. Breve descripción (en castellano):

- Diseño modular y creación de bibliotecas.
- Herramientas de depuración, pruebas y validación.
- Gestión de errores.
- Proyectos informáticos de programación.
- Orientación a Objetos.
- Polimorfismo, Herencia, Objetos complejos
- Persistencia de los objetos.
- Patrones para la resolución de problemas.
- Programación orientada a aspectos.

### 1.2. Breve descripción (en inglés):

- Modular Design
- Debug tools. Testing and validation
- Exception Management
- Object oriented Concepts
- Object Persistancy
- Introduction to Patterns
- Aspect Oriented Programing

## 2. Situación de la asignatura

### 2.1. Contexto dentro de la titulación:

Asignatura obligatoria de 2º de Grado en Ingeniería Informática (primer cuatrimestre) adscrita al Departamento de Tecnologías de la Información en el Área de Ciencias de la Computación e Inteligencia Artificial.

Esta asignatura requiere de los conocimientos adquiridos en las asignaturas de primer curso:

- Fundamentos de Programación
- Fundamentos de Análisis de Algoritmos
- Estructuras de Datos I

Finalmente, los conocimientos adquiridos en esta asignatura sirven de base para las asignaturas de segundo y tercer curso:

- Estructuras de Datos II
- Principios y Fundamentos de Ingeniería del Software
- Programación Concurrente y Distribuida

### 2.2. Recomendaciones:

Se recomienda haber cursado las asignaturas de primer curso indicadas en el apartado anterior antes de comenzar esta asignatura.

Debido al marcado carácter práctico de la asignatura, se recomienda que el alumnado realice la mayor parte de los ejercicios propuestos por el profesor en las sesiones de teoría, así como probar todo código mostrado en pizarra y/o transparencias en dichas sesiones.

## 3. Objetivos (Expresados como resultados del aprendizaje):

Al finalizar el curso, el alumnado debe haber adquirido los conocimientos y habilidades necesarias para:

- Desarrollar una aproximación disciplinada a la especificación, implementación, verificación y documentación de programas.
- Apreciar el papel central que juega la abstracción en la tarea de programar.
- Conocer estructuras de datos, algoritmos y esquemas de uso general.
- Proporcionar los fundamentos teóricos y prácticos básicos para cursar posteriores estudios en programación.
- Introducir el paradigma de la programación orientada a objetos.
- Aprender lenguajes de programación orientados a objetos.

#### 4. Competencias a adquirir por los estudiantes

##### 4.1. Competencias específicas:

- **CC05:** Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.
- **CC06:** Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos
- **CC07:** Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema
- **CC08:** Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados

##### 4.2. Competencias básicas, generales o transversales:

- **CB1:** Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio
- **CB5:** Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía
- **CT2:** Desarrollo de una actitud crítica en relación con la capacidad de análisis y síntesis.
- **CT3:** Desarrollo de una actitud de indagación que permita la revisión y avance permanente del conocimiento.
- **CT4:** Capacidad de utilizar las Competencias Informáticas e Informacionales (CI2) en la práctica profesional.

## 5. Actividades Formativas y Metodologías Docentes

### 5.1. Actividades formativas:

- Sesiones de Teoría sobre los contenidos del Programa.
- Sesiones de Resolución de Problemas.
- Sesiones Prácticas en Laboratorios Especializados o en Aulas de Informática.
- Actividades Académicamente Dirigidas por el Profesorado: seminarios, conferencias, desarrollo de trabajos, debates, tutorías colectivas, actividades de evaluación y autoevaluación.

### 5.2. Metodologías docentes:

- Clase Magistral Participativa.
- Desarrollo de Prácticas en Laboratorios Especializados o Aulas de Informática en grupos reducidos.
- Resolución de Problemas y Ejercicios Prácticos.
- Tutorías Individuales o Colectivas. Interacción directa profesorado-estudiantes.
- Planteamiento, Realización, Tutorización y Presentación de Trabajos.
- Conferencias y Seminarios.
- Evaluaciones y Exámenes.

### 5.3. Desarrollo y justificación:

#### **Clases teóricas y de problemas en el aula**

Las clases de teoría se impartirán combinando clases magistrales y problemas al total de alumnos del grupo de teoría. Durante estas sesiones el alumno adquirirá los conceptos teóricos, apoyados con ejemplos y ejercicios aclaratorios, que le capacitarán para alcanzar las competencias establecidas en el apartado correspondiente de competencias de esta guía. Los problemas planteados durante las sesiones de teoría serán resueltos por el profesor y/o por los alumnos de forma individual o en grupo.

Para el desarrollo de las clases se utilizarán transparencias apoyadas con el uso de la pizarra cuando sea necesario y ordenador portátil con retroproyector para la compilación y demostración de programas.

La asistencia a las sesiones de teoría no es obligatoria, aunque si es recomendable.

#### **Clases de Laboratorio**

Las clases prácticas se impartirán en el laboratorio utilizando un ordenador personal para cada alumno, del total de grupo de prácticas. En estas sesiones cada alumno de forma individual deberá implementar diferentes problemas utilizando Java y/o C++ como lenguaje de programación. El objetivo de estas sesiones es enfrentar al alumno con un desarrollo práctico que le permitirá adquirir las competencias procedimentales específicas, así como la mayoría de las competencias transversales.

La asistencia a las sesiones de laboratorio es obligatoria (se debe tener una asistencia mínima del 80% a prácticas para poder ser evaluada).

#### **Exposiciones y Seminarios**

Para que los alumnos conozcan aspectos avanzados de lenguajes de programación OOP se podrán impartir seminarios sobre las alternativas existentes.

La asistencia a dichos seminarios no es obligatoria, aunque si es recomendable.

## 6. Temario desarrollado:

### Tema 0: Introducción Conceptos Avanzados de Orientación a Objetos

- 0.1 Herencia.
- 0.2 Visibilidad.
- 0.3 Polimorfismo y otros conceptos.

### Tema 1: Herencia y Polimorfismo en C++

- 1.1 Herencia simple vs múltiple. Redefinición de métodos.
- 1.2 Jerarquías de clases y visibilidad.
- 1.3 Polimorfismo y ligadura dinámica: métodos virtuales.
- 1.4 Tipos de constructores. Destructores polimórficos. Funciones amigas.
- 1.5 Excepciones OOP.
- 1.6 Manejo de ficheros OOP.
- 1.7 Genéricos.

### Tema 2: Herencia y Polimorfismo en Java

- 2.1 Herencia.
- 2.2 Jerarquías de clases y visibilidad.
- 2.3 Polimorfismo.
- 2.4 Tipos de constructores y destructores polimórficos.
- 2.5 Excepciones OOP.
- 2.6 Manejo de ficheros OOP.
- 2.7 Genéricos.

### Tema 3: Patrones simples en Java

- 3.1 Diseño de tipos.
- 3.2 Contratos, implementación y reutilización.
- 3.3 Factorías e iterables. Colecciones.

### Tema 4: Librerías y herramientas en Java

- 4.1 Librerías gráficas.
- 4.2 Generación de documentación.
- 4.3 Generación de archivos Jar.

### Seminarios:

Otras Técnicas/Lenguajes/Plataformas orientados a objetos.

## 7. Bibliografía

### 7.1. Bibliografía básica:

#### **C++**

Stroustrup, Bjarne. El Lenguaje de Programación C++. Ed: Addison Wesley [2002]  
Deitel, Harvey M. -; Deitel, Paul J. Cómo Programar en C++ (9ª ed.). Ed: Pearson Educación, [2014]  
Cohoon, James P. - Davidson, Jack W. Programación y Diseño en C++. Ed: McGraw-Hill [2000]

#### **Java**

Eckel, Bruce. Piensa en Java (4ª ed.). Ed: Pearson-Prentice Hall, [2007]  
Deitel, Paul J. - Deitel, Harvey M. Cómo Programar en Java (9ª ed.). Ed: Pearson Educación, [2012]  
Arnold, Ken - Gosling, J. - Colmes, D. El Lenguaje de Programación Java. Ed: Pearson Educación, [2001]

### 7.2. Bibliografía complementaria:

Weiss, Mark Allen. Estructuras de Datos en JAVA (4ª ed.). Ed: Pearson Educación, [2013]  
Aho, Alfred V. - Hopcroft, J. - Ullman, J. D. Estructuras de Datos y Algoritmos. Ed: Addison Wesley. [1988]

## 8. Sistemas y criterios de evaluación.

### 8.1. Sistemas de evaluación:

- Examen de teoría/problemas
- Defensa de Trabajos e Informes Escritos
- Examen de prácticas

### 8.2. Criterios de evaluación y calificación:

## Criterios de Evaluación

En función de sus circunstancias personales, los alumnos podrán acogerse a la modalidad de **Evaluación Continua** (por defecto) o **Evaluación única final**.

Los alumnos que quieran acogerse a la modalidad de **Evaluación única final** deberán presentar (en las dos primeras semanas de impartición de la asignatura, o en las dos semanas siguientes a su matriculación si ésta se ha producido con posterioridad al inicio de la asignatura) una solicitud en el REGISTRO GENERAL de la Universidad, en cualquiera de sus REGISTROS AUXILIARES o en el REGISTRO TELEMÁTICO, dirigida a la dirección del departamento y al coordinador de la asignatura.

Además, los alumnos que se presenten a convocatorias especiales (convocatoria ordinaria III y/o a la convocatoria **Extraordinaria**) serán evaluados mediante la modalidad de **Evaluación única final**.

## Evaluación Continua

Los principios de evaluación de la asignatura siguen unos criterios de **evaluación** preferentemente **continua**, entendiéndose por tal la evaluación diversificada que se lleva a cabo en distintos momentos del curso académico en curso. Esta evaluación diversificada, sólo es aplicable a convocatorias ordinarias I y II, y se realiza mediante los siguientes sistemas de evaluación y ponderaciones:

### Convocatoria ordinaria I:

Aspecto.....	Criterio.....	Instrumento.....	Peso.....
Conocimientos Teóricos.....	<i>Exposición escrita de conocimientos.....y su aplicación a distintas situaciones</i>	Examen de Teoría Escrito.....	60%
Conocimientos Prácticos.....	<i>Capacidad de implementar programas.....en C++ y Java</i>	Examen de Prácticas reducido en C++ y Java.....	30%
Trabajos/Informes escritos....	<i>Calidad de los trabajos prácticos.....</i>	Entrega y Defensa de Prácticas.....	10%

### Criterios de Evaluación de las Competencias adquiridas en la asignatura:

- Examen de teoría (CC06, CC07, CB1, CB5, CT2, CT3, CT4)
- Examen de prácticas reducido (CC05, CC07, CC08, CB1, CB5, CT2, CT3, CT4)
- Trabajos e Informes escritos (CC05, CC06, CC07, CC08, CB1, CB5, CT2, CT3, CT4)

### Convocatoria ordinaria II:

Aspecto.....	Criterio.....	Instrumento.....	Peso.....
Conocimientos Teóricos.....	<i>Exposición escrita de conocimientos.....y su aplicación a distintas situaciones</i>	Examen de Teoría Escrito.....	60%
Conocimientos Prácticos.....	<i>Capacidad de implementar programas.....en C++ y Java</i>	Examen de Prácticas completo en C++ o Java.....	40%

### Criterios de Evaluación de las Competencias adquiridas en la asignatura:

- Examen de teoría (CC06, CC07, CB1, CB5, CT2, CT3, CT4)
- Examen de prácticas completo (CC05, CC06, CC07, CC08, CB1, CB5, CT2, CT3, CT4)

## Evaluación parte Teórica (Peso 60%)

### Convocatorias ordinarias I y II:

- El examen Teórico consistirá en una prueba escrita compuesta por problemas prácticos y opcionalmente un test y/o preguntas cortas de teoría.
- En el caso de existir test cada pregunta acertada se puntuará positivamente y cada pregunta errónea negativamente de forma proporcional al número de respuestas, no puntuando las no contestadas.
- El examen podrá incluir preguntas y/o problemas sobre los seminarios, actividades de laboratorio y actividades propuestas por el profesor durante el curso.
- El examen tiene carácter presencial e individual y una duración de 2 horas.

## Evaluación parte Práctica (Peso 40%)

### Convocatoria ordinaria I:

#### - Defensa de Trabajos e Informes (10%)

- \* El profesor evaluará la calidad de los trabajos prácticos, considerando la solución de código propuesta y la claridad y eficiencia de la implementación desarrollada.
- \* Los trabajos prácticos (2 Actividades Académicamente Dirigidas) se deben entregar en fecha y forma, para poder ser evaluados.

#### - Examen de prácticas reducido (30%)

- \* Se realizará dos exámenes de modificación reducido sobre cada uno de los trabajos prácticos (uno a mitad de cuatrimestre y otro al final del cuatrimestre) propuestos a lo largo del curso. Cada examen se realizará de forma presencial e individual en una sesión de prácticas y tendrá una duración de 1.5 horas.

\* Aquellos alumnos que **no entreguen las prácticas no podrán realizar el examen de Prácticas reducido.**

\* Tener una asistencia mínima del 80% a prácticas y presentarse a los 2 exámenes de modificación (examen de Prácticas reducido).

#### **Convocatoria ordinaria II:**

##### **- Defensa de Trabajos e Informes (0%)**

\* El valor de estas actividades NO se guardará para esta convocatoria ya que son actividades realizadas durante el periodo de docencia donde el alumnado tiene que entregar las prácticas propuestas.

##### **- Examen de prácticas completo (40%)**

- Existirá un examen final de prácticas (examen de Prácticas completo consistente en desarrollar un programa de la complejidad de los realizados en las prácticas propuesta a lo largo del curso) para aquellos alumnos que tengan el examen de Prácticas reducido suspenso (calificación < 5 sobre 10) o que no lo hayan podido realizar por no entregar en la convocatoria ordinaria I las prácticas propuestas.

- El examen se realizará en ordenador en un aula de informática a continuación del examen de la parte teórica, el mismo día.

- El examen tiene carácter presencial e individual y una duración de 2.5 horas.

#### **Calificación:**

- La nota final se obtiene ponderando la media aritmética de las calificaciones de la parte teórica (60%) y de la parte prácticas (40%).
- La NOTA FINAL será:

#### **Convocatoria ordinaria I:**

Nota Acta = Examen Teoría x 60% + Examen Prácticas x 30% + Defensa Trabajos e Informes x 10%

#### **Convocatoria ordinaria II:**

Nota Acta = Examen Teoría x 60% + Examen Prácticas x 40%

**Para aplicar la fórmula anterior es necesario tener aprobado el examen Teórico y Práctico (calificación  $\geq 5$  sobre 10) En caso contrario el alumno tendrá SUSPENSA la asignatura y la nota en ACTAS será la menor de las siguientes (4.50, Nota Acta)**

## **Evaluación única final**

La evaluación única final consistirá, **PARA TODAS LAS CONVOCATORIAS**, en un solo acto académico que estará formado por las siguientes pruebas:

##### **• Evaluación Parte Teórica.(60%)**

- Se evalúa mediante un examen Teórico-Práctico consistente en una prueba escrita compuesta por problemas prácticos y opcionalmente un test y/o preguntas cortas de teoría.

- En el caso de existir test cada pregunta acertada se puntuará positivamente y cada pregunta errónea negativamente de forma proporcional al número de respuestas, no puntuando las no contestadas.

- El examen podrá incluir preguntas y/o problemas sobre los seminarios, actividades de laboratorio y actividades propuestas por el profesor durante el curso.

- El examen tiene carácter presencial e individual y una duración de 2 horas.

##### **• Evaluación Parte Práctica.(40%)**

- Se evalúa mediante un examen de Prácticas completo consistente en desarrollar un programa de la complejidad de los realizados en las prácticas propuesta a lo largo del curso.

- El examen se realizará en ordenador en un aula de informática a continuación del examen de la parte teórica, el mismo día.

- El examen tiene carácter presencial e individual y una duración de 2.5 horas.

#### **Criterios de Evaluación de las Competencias adquiridas en la asignatura:**

- Examen de teoría (CC06, CC07, CB1, CB5, CT2, CT3, CT4)
- Examen de prácticas completo (CC05, CC06, CC07, CC08, CB1, CB5, CT2, CT3, CT4)

#### **Calificación:**

- La nota final se obtiene ponderando la media aritmética de las calificaciones del examen teórico (60%) y del examen práctico (40%).
- La NOTA FINAL será:

**Nota Acta** = Examen Teoría x 60% + Examen Prácticas x 40%

**Para aplicar la fórmula anterior es necesario tener aprobado el examen Teórico y Práctico (calificación  $\geq 5$  sobre 10) En caso contrario el alumno tendrá SUSPENSA la asignatura y la nota en ACTAS será la menor de las siguientes (4.50, Nota Acta)**

## Avisos Importantes:

1. Los alumnos con **Evaluación Continua** que tengan un número de faltas de asistencia no justificada a prácticas superior al 20% o que no entreguen las prácticas en el plazo requerido, tendrán la calificación de **Suspenso** en la parte práctica de la asignatura, teniendo por lo tanto, que presentarse al Examen Final de prácticas de la convocatoria ordinaria II.
2. De forma excepcional, aquellos alumnos con **Evaluación Continua** que no puedan asistir a las prácticas por motivos laborales y siempre que acrediten debidamente dicha situación, estarán exentos de venir a prácticas pero tendrán que entregar las prácticas en el mismo plazo requerido que el resto de estudiantes (la justificación de su no asistencia a prácticas no le exime de presentarlas en el mismo plazo que el resto de estudiantes). En cualquier caso, los alumnos acogidos a esta modalidad semipresencial deben realizar el examen de modificación (2 a lo largo del cuatrimestre) el mismo día que el resto de compañeros del grupo de prácticas donde estén apuntados.
3. Los estudiantes que no puedan asistir a clase podrán solicitar una **Evaluación única final** en los términos indicados en el REGLAMENTO DE EVALUACIÓN PARA LAS TITULACIONES DE GRADO Y MÁSTER OFICIAL DE LA UNIVERSIDAD DE HUELVA. Esto implicará la renuncia expresa a la evaluación continua, sin posibilidad de que el estudiante pueda cambiar de sistema.
4. Los alumnos que se presenten a convocatorias especiales (convocatoria ordinaria III y/o a la convocatoria **Extraordinaria**) serán evaluados mediante la modalidad de **Evaluación única final**.
5. Aquellos estudiantes que no puedan asistir al examen por las causas justificadas e indicadas en el REGLAMENTO DE EVALUACIÓN PARA LAS TITULACIONES DE GRADO Y MÁSTER OFICIAL DE LA UNIVERSIDAD DE HUELVA, tendrán que solicitar un **examen de incidencia**.

## Nota sobre la calificación “Matrícula de Honor”:

La mención de **Matrícula de Honor** podrá ser otorgada a alumnos que hayan obtenido una calificación igual o superior a 9,0, si así lo solicitan. En ningún caso el número de **Matrículas de Honor** concedidas será superior al máximo establecido para la asignatura en el curso académico en curso. En el caso de haber más candidatos que posibilidades de **Matrículas de Honor** por número de estudiantes en la asignatura, y con el objetivo de discriminar situaciones de equidad en la calificación final, se seguirán los siguientes criterios:

- 1º Mejor nota en examen de teoría.
- 2º Mejor nota en examen de práctica (para los alumnos con evaluación continua mejor nota media de los dos exámenes reducidos y en caso de empate nota más alta en alguno de los dos exámenes reducidos; para los alumnos con evaluación única final nota más alta del examen práctico).
- En caso de seguir el empate la **Matrícula de Honor** se asignará basándose en la mayor participación en clase e implicación del alumno en la asignatura (en todo tipo de sesiones), a juicio de los profesores de la asignatura.

## Criterio para guardar las notas para convocatorias posteriores a la I:

### Defensa de Trabajos e Informes:

- La nota de la Defensa de Trabajos e Informes NO se guardan para convocatorias ordinarias posteriores a la I, ya que son actividades realizadas durante el periodo de docencia donde el alumnado tiene que participar activamente.

### Examen de Teoría:

- La nota del examen de Teoría se guarda hasta la convocatoria ordinaria II inclusive, siempre y cuando el alumno tenga una calificación  $\geq 5$  (sobre 10), no guardándose para convocatorias posteriores (convocatoria ordinaria III y convocatoria **extraordinaria**).

### Examen de Prácticas:

- La nota del examen de Prácticas se guarda hasta la convocatoria ordinaria II inclusive, siempre y cuando el alumno tenga una calificación  $\geq 5$  (sobre 10), no guardándose para convocatorias posteriores (convocatoria ordinaria III y convocatoria **extraordinaria**).
- En ese caso, la nota considerada obtenida en la convocatoria ordinaria II vendrá dada por la siguiente fórmula:  
**(Examen de Prácticas x 30% + Entrega y Defensa de Trabajos e Informes x 10%) / 40% (pondera un 40%)**

**En cualquier caso, ninguna nota (teórica, práctica y defensa de Trabajos e Informes) de un curso académico se guarda para cursos académicos posteriores.**

### 9. Organización docente semanal orientativa:

	Semanas	Grupos Grandes	Grupos Reducidos Aula Estándar	Grupos Reducidos Aula de Informática	Grupos Reducidos Laboratorio	Grupos Reducidos prácticas de campo	Pruebas y/o actividades evaluables	Contenido desarrollado
#1	2	0	2	0	0		Presentación. Tema 0	
#2	2	0	2	0	0		Tema 1	
#3	2	0	2	0	0		Tema 1	
#4	2	0	2	0	0		Tema 1	
#5	2	0	2	0	0		Tema 1	
#6	2	0	2	0	0		Tema 1	
#7	2	0	2	0	0	Sesión de Problemas	Tema 1	
#8	2	0	2	0	0		Tema 2	
#9	2	0	2	0	0	Entrega AAD C++	Tema 2	
#10	2	0	2	0	0	Prueba Laboratorio C++	Tema 2	
#11	2	0	2	0	0		Tema 2	
#12	2	0	2	0	0		Tema 3	
#13	2	0	2	0	0	Sesión de Problemas	Tema 3	
#14	2	0	2	0	0	Entrega AAD Java	Tema 3	
#15	2	0	2	0	0	Prueba Laboratorio Java	Tema 4	
	30	0	30	0	0			