



**UNIVERSIDAD DE CÓRDOBA**  
**SISTEMA DE CRÉDITOS EUROPEOS ECTS**  
**GUÍA DOCENTE DE ASIGNATURA**



- **DATOS BÁSICOS DE LA ASIGNATURA**
  - NOMBRE: **Lenguajes de Inteligencia Artificial**
  - CÓDIGO DE LA ASIGNATURA: **6230036**
  - TIPO (troncal / obligatoria / optativa): **Optativa**
  - Créditos totales LRU / ECTS: **4'5 / 4**
  - Créditos LRU / ECTS teóricos: **3 / 2'5**
  - Créditos LRU / ECTS prácticos: **1'5 / 1'5**
  - CICLO: **1º**
  - CURSO: **2º**
  - CUATRIMESTRE: **1º**
  - TITULACIÓN: **Ingeniería Técnica en Informática de Sistemas**
  - AÑO DE PLAN DE ESTUDIOS: **1999**
  - CURSO ACADÉMICO: **2009 – 2010**
  
- **DATOS BÁSICOS DE LOS PROFESORES**
  - NOMBRE: Nicolás Luis Fernández García
  - CENTRO: Escuela Politécnica Superior
  - DEPARTAMENTO: Informática y Análisis Numérico
  - ÁREA DE CONOCIMIENTO: Ciencia de la Computación e Inteligencia Artificial
  - DESPACHO: Edificio Einstein, tercera planta, pasillo sur. Campus de Rabanales
  - CORREO ELECTRÓNICO: [ma1fegan@uco.es](mailto:ma1fegan@uco.es)
  - TELÉFONO: 957 21 83 47
  - URL WEB: <http://www.uco.es/users/ma1fegan/>
  
- **DATOS ESPECÍFICOS DE LA ASIGNATURA**
  1. **DESCRIPTORES:**
    - Introducción a la inteligencia artificial
    - Programación funcional
    - Programación lógica
  
  2. **SITUACIÓN**
    - 2.1. **PRERREQUISITOS**
      - No se establece ningún prerrequisito.
  
    - 2.2. **CONTEXTO DENTRO DE LA TITULACIÓN**
      - Está asignatura está relacionada con las siguientes asignaturas:
        - Metodología y tecnología de la programación (primer curso)

- Matemáticas I (primer curso)
- Matemática II (primer curso)
- Estructura de datos y la información (segundo curso)
- Informática aplicada (segundo curso)

### **2.3. RECOMENDACIONES**

- Se deben tener conocimientos sobre los siguientes conceptos:
  - Diseño de algoritmos: explicados en la asignatura de primer curso denominada “Metodología y Tecnología de la Programación”.
  - Lógica matemática: explicados en la asignatura de primer curso denominada “Matemáticas II”.

## **3. COMPETENCIAS**

### **3.1. COMPETENCIAS TRANSVERSALES / GENÉRICAS**

- Aprendizaje autónomo
- Capacidad de análisis y síntesis
- Capacidad de aplicar los conocimientos a la práctica
- Resolución de problemas
- Trabajo individual y en grupo
- Comunicación oral y escrita

### **3.2. COMPETENCIAS ESPECÍFICAS**

- Cognitivas (Saber):
  - Idioma: español e inglés
  - Matemáticas
  - Nuevas tecnologías TIC
  - Conocimientos de informática
- Procedimentales / Instrumentales (Saber hacer):
  - Redacción en interpretación de documentación técnica
  - Estimación y programación del trabajo
  - Planificación, organización y estrategia
- Actitudinales (Ser):
  - Calidad
  - Toma de decisión
  - Capacidad de iniciativa y participación

## **4. OBJETIVOS**

- Describir los conceptos principales de la programación funcional y de la programación lógica

- Explicar las características básicas y fundamentales de un lenguaje de programación funcional (Scheme) y de un lenguaje de programación lógica (Prolog).

## 5. METODOLOGÍA

5.1. Número total de horas de trabajo del alumno: 100 horas

- 45 horas presenciales
- 55 horas no presenciales

5.2. Horas presenciales: 45 horas.

- **Clases de Teoría:** 30 horas
  - Explicaciones del profesor: 27'5 horas
  - Encuesta de evaluación del profesorado: 0'5 horas
  - **Actividades supervisadas:** 2 horas
    - Exposiciones de trabajos de teoría: 2 horas
- **Clases de Prácticas:** 15 horas
  - Explicaciones del profesor: 2 horas.
    - Explicación de las herramientas de software: 2 horas
  - **Actividades supervisadas:** 13 horas
    - Elaboración de las prácticas: 9 horas
    - Corrección de las prácticas: 4 horas

**Tabla 1.- Distribución de las 45 horas presenciales**

	<b>Explicaciones del Profesor</b>	<b>Actividades Académicas Supervisadas</b>	<b>Encuesta de evaluación del profesor</b>	<b>Total</b>
<b>Teoría</b>	<b>27,5</b>	<b>2</b>	<b>0'5</b>	<b>30</b>
<b>Prácticas</b>	<b>2</b>	<b>13</b>		<b>15</b>
<b>Total</b>	<b>29,5</b>	<b>15</b>	<b>0,5</b>	<b>45</b>

5.3. Horas **no** presenciales: 55 horas

- Estudio de los temas de teoría: 20 horas
- Elaboración de las prácticas: 20 horas
- Preparación de trabajos de teoría: 6 horas
- Preparación del examen escrito: 6 horas
- Realización de examen escrito: 3 horas

## 6. TÉCNICAS DOCENTES

- Las técnicas docentes que se van a utilizar son:
  - Clases de teoría
  - Elaboración y exposición de trabajos de teoría
  - Clases de prácticas
  - Corrección de las prácticas

- Tutorías individualizadas

## 6.1. DESARROLLO Y JUSTIFICACIÓN

- Clases de teoría:
  - Se hará una reseña inicial del contenido de cada tema y se indicará su relación con los otros temas.
  - Al comenzar la explicación de una sección de un tema, se indicarán las relaciones que posee con otras secciones del mismo tema o de temas diferentes.
  - Se explicará detenidamente cada sección de cada tema teórico.
  - Se ilustrarán con ejemplos las aplicaciones prácticas de los fundamentos teóricos explicados en cada sección.
  - Durante la explicación de cada tema, se propondrán cuestiones relativamente simples a los alumnos para estimular el aprendizaje de los conceptos teóricos.
  - Se utilizará la pizarra para el desarrollo de los contenidos teóricos y el proyector de transparencias para mostrar algunos algoritmos, gráficos, tablas y ejemplos.
  - Se entregará a los alumnos fotocopias de los algoritmos, gráficos, tablas y ejemplos mostrados con el proyector de transparencias.
  - Se entregarán a los alumnos hojas de ejercicios de los temas de teoría.
  - Al finalizar la explicación de cada tema, se hará un breve resumen para resaltar los conceptos más importantes.
  - Todo el material docente entregado en las clases de teoría o prácticas estará disponible en la página web de la asignatura:  
**<http://www.uco.es/users/ma1fegan/>**
  - Se fomentará la utilización oral y escrita de la lengua inglesa.
- Elaboración y exposición de trabajos de teoría:
  - El profesor propondrá los trabajos de teoría que los alumnos deberán preparar y exponer a lo largo del curso.
  - Los trabajos podrán hacerse en grupos de un máximo de tres personas.
  - Antes de exponer el trabajo, los alumnos deberán reunirse con el profesor para comprobar que el trabajo elaborado es correcto y evaluarlo.
  - Si varios grupos de alumnos hubieran hecho de manera independiente el mismo el trabajo, el profesor elegirá el mejor de ellos para que sea expuesto en público.
  - El día de la exposición del trabajo, el profesor hará una presentación inicial y después continuarán los alumnos que hayan elaborado el trabajo correspondiente.
- Elaboración de las prácticas:

- Se explicarán los recursos informáticos necesarios para el desarrollo de las prácticas: intérpretes (drscheme y swi-prolog), editores de texto, entorno gráfico integrado, etc.
  - Se describirán de los objetivos que se pretenden conseguir con la elaboración de cada una de las prácticas.
  - Se utilizará la pizarra para el desarrollo de los fundamentos prácticos y el proyector de transparencias y el material informático (hardware: ordenadores; software: drscheme, swi-prolog, editores de texto, etc.) para desarrollar los ejemplos.
  - Se entregarán a los alumnos fotocopias o ficheros digitales de manuales de los lenguajes Scheme y Prolog, ejemplos prácticos y los enunciados de las prácticas.
  - Los alumnos desarrollarán las prácticas codificando y documentando los programas.
  - Los alumnos realizarán tres prácticas de Scheme y una de Prolog.
  - Se fomentará la utilización de la lengua inglesa mediante el empleo de material escrito en dicha lengua: hojas de ejercicios, ejemplos, manuales.
- Corrección de las prácticas:
    - El profesor corregirá y evaluará en presencia del alumno los trabajos de prácticas que haya realizado.
    - El profesor indicará al alumno los posibles fallos y propondrá posibles soluciones alternativas.
  - Tutorías individualizadas:
    - Los alumnos deben utilizar estas tutorías a lo largo de todo el curso y no sólo antes de la fecha del examen.
    - El profesor intentará resolver las dudas particulares que pueda tener cada alumno en relación con los temas de teoría, los trabajos de las exposiciones, las prácticas, etc.
    - Aunque las dudas más simples puedan plantearse mediante correo electrónico, es preferible que haya una reunión del profesor y el alumno para resolver las dudas más complejas.
  - Objetivo transversal: se fomentará la utilización de la lengua inglesa en las actividades docentes.

## 7. BLOQUES TEMÁTICOS

- Programación funcional: Scheme
- Programación lógica: Prolog

## 8. BIBLIOGRAFÍA

### 8.1. GENERAL

#### Lenguaje Scheme

- Abelson, H., Sussman, G. J. y Sussman, J. “Structure and Interpretations of Computers Programs”. Second edition. The MIT Electrical Engineering and Computers Science Series, 1996. ISBN: 0-262-01153-0.
- Kelsey, R., Clinger, W, Rees, J. y otros: “Revised<sup>5</sup> Report on the Algorithmic Language Scheme”, 1998.  
<http://www.uco.es/users/ma1fegan/manuales/lia/r5rs.pdf>

#### Lenguaje Prolog

- Mellish, C. S., Cloksin, W. F. “Programming in PROLOG”. Springer –Verlag, 2003, ISBN: 978-3-540-00678-7.
- Shapiro, E. y Sterling, L. “The Art of Prolog”. Second edition. MIT Press, 1994. ISBN: 0-262-19338-8.

### 8.2. ESPECÍFICA

#### Lenguaje Scheme

- Abelson, H., Sussman, G. J. y Sussman, J. “Structure and Interpretations of Computers Programs”. The MIT Electrical Engineering and Computers Science Series, 1993. ISBN: 0-262-01077-1.
- Dybvig, R. Kent. “The Scheme Programming Language” Second Edition. Ansi Scheme. Prentice Hall, 1996. ISBN: 0-13-454646-6.
- Friedman, D. P. y Felleisen, M. “The Seasoned Schemer”. MIT Press, 1996 ISBN: 0-262-56100-X.
- Grilmeyer, O. “Exploring Computer Science with Scheme”. Springer Verlag, 1999. ISBN: 0-387-94895-3.
- Harvey, B. y Wright, M. “Simply Scheme: Introducing Computer Science”. The MIT Press, 1994. ISBN: 0-262-08226-8.
- Pearce, Jon. “Programming and Meta-Programming in Scheme”. Springer –Verlag, 1998. ISBN: 0-387-98320-1.
- Waston, Mark. “Programming in Scheme. Learn Scheme Through Artificial Intelligence Programs”. Springer - Verlag, 1996. ISBN: 0-387-94681-0.

#### Lenguaje Prolog

- Bramer, M. “Logic Programming with Prolog”. Springer, 2005. ISBN-10: 1-85233-938-2. ISBN-13: 978-1852-33938-8.
- Bratko, I. “PROLOG: Programming for Artificial Intelligence”, Segunda edición. Addison-Wesley, 1994. ISBN: 0-201-41606-9.
- Giannesini, F.; Kanoui, H; Pasero, R. y van Caneghem, M. “Prolog”. Addison Wesley Iberoamericana, 1989. ISBN: 84-7829-003-6.

- Mellish, C. S., Cloksin, W.F. “Programming in PROLOG”. Springer - Verlag, 1994. ISBN: 3-540-58350-5.
- Mellish, C. S., Cloksin, W.F. “Programación en PROLOG”. Editorial Gustavo Gili, S. A., 1987. ISBN: 84-252-1339-8.

## 9. TÉCNICAS DE EVALUACIÓN

- La evaluación de la asignatura estará basada en
  - Trabajos de teoría
  - Trabajos de prácticas
  - Examen escrito de teoría
- La “calificación final” de la asignatura se obtendrá aplicando la siguiente expresión:

$$\begin{aligned} \text{Calificación final} = & 1/6 \text{ Trabajo de teoría} \\ & + 1/2 \text{ Examen escrito de teoría} \\ & + 1/3 \text{ Trabajos de prácticas} \end{aligned}$$

- Para poder aprobar la asignatura es necesario obtener “una calificación final” superior o igual a cinco (5) puntos, exigiéndose, además, que la nota del examen escrito de teoría sea igual o superior a cuatro (4) puntos.
- La nota del trabajo de teoría se obtendrá al partir de la documentación elaborada y de la exposición realizada en la clase, si ésta se produce.
- La nota de prácticas se obtendrá a partir de los trabajos correctamente realizados.
- Se establecerá una fecha máxima para la entrega de las prácticas y de los trabajos de teoría.
- **Nota importante:** en las convocatorias extraordinarias de septiembre, diciembre o enero, la evaluación de la asignatura se realizará exclusivamente mediante un examen final de teoría, que, obviamente, representará el 100% de la calificación global de la asignatura.

## 10. ORGANIZACIÓN DOCENTE SEMANAL

SEMANA	TEORÍA			PRÁCTICAS			Examen	Actividad docente
	Explicación del profesor	Actividad Académica dirigida: Exposiciones de los alumnos	Actividad Académica dirigida: Encuesta de evaluación del profesorado	Explicación del profesor	Actividad Académica dirigida: Elaboración de prácticas	Actividad Académica dirigida: Corrección de prácticas		
1ª semana	2			1				Presentación de la asignatura Explicación de los recursos de software
2ª semana	2				1			Tema 1 Práctica 1
3ª semana	2				1			Tema 2 Práctica 1
4ª semana	2				1			Tema 2 Práctica 1
5ª semana	2				1			Tema 3 Práctica 1
6ª semana	2					1		Tema 4 Corrección de Práctica 1
7ª semana	2				1			Tema 5 Práctica 2
8ª semana	2				1			Tema 6 Práctica 2
9ª semana	2					1		Temas 7 y 8 Corrección de la práctica 2
10ª semana	2				1			Tema 9 Práctica 3



SEMANA	TEORÍA			PRÁCTICAS			Examen	Actividad docente
	Explicación del profesor	Actividad Académica dirigida: Exposiciones de los alumnos	Actividad Académica dirigida: Encuesta de evaluación del profesorado	Explicación del profesor	Actividad Académica dirigida: Elaboración de prácticas	Actividad Académica dirigida: Corrección de prácticas		
11ª semana	2				1			Temas 9 y 10 Práctica 3
12ª semana	2					1		Tema 10 Corrección de la práctica 3
13ª semana	2			1				Tema 11 Explicación de los recursos de software
<b>VACACIONES DE NAVIDAD</b>								
14ª semana	1'5		0'5		1			Tema 12 (* Encuesta de evaluación de profesorado Práctica 4
15ª semana		2				1		Exposición de los alumnos Corrección de la práctica 4
<b>PERIODO DE EXÁMENES</b>								
							3	(* Examen escrito
<b>SUBTOTALES</b>	<b>27'5</b>	<b>2</b>	<b>0'5</b>	<b>2</b>	<b>9</b>	<b>4</b>	<b>3</b>	
<b>TOTAL</b>	<b>30</b>			<b>15</b>				

**Nota:** (\*) No se conocen todavía las fechas de la encuesta de evaluación del profesorado y del examen final

## 11. TEMARIO DESARROLLADO

### Primera parte: Scheme

1. Introducción al lenguaje Scheme
  - Características fundamentales de la programación funcional
  - Reseña histórica de Scheme
    - Lisp
    - Compilación e interpretación
    - Ámbito dinámico y ámbito léxico o estático
    - Origen de Scheme
  
2. Expresiones y Funciones
  - Elementos básicos de Scheme
    - Caracteres
    - Cadenas de caracteres
    - Números
    - Identificadores: variables y palabras reservadas
  - Expresiones
    - Literales
    - Variables
    - Operadores aritméticos
  - Definición de función
  - Reglas de ámbito léxico
    - Definiciones internas
    - Estructuras de bloques
  - Funciones anónimas: la forma especial “lambda”
  - Las formas especiales “let” y “let\*”
  
3. Predicados y sentencias condicionales
  - Operadores relacionales
  - Operadores lógicos
  - Predicados primitivos
    - Predicados simbólicos
    - Predicados numéricos
    - Predicados de equivalencia
  - Formas especiales condicionales:
    - Forma especial “if”
    - Forma especial “cond”
    - Forma especial “case”
  
4. Iteración y recursión
  - Forma especial iterativa “do”
  - Recursión simple
  - Recursión múltiple
  - Recursión de cola
  - Forma especial “let con nombre”

- Funciones utilizadas como parámetros
  - Funciones devueltas como resultados
5. Tipos compuestos de datos
- Vector
  - Operaciones con vectores
  - Pares
  - Listas
  - Operaciones con listas
  - Conversión de listas y vectores
  - Funciones con argumentos opcionales
6. Abstracción de datos
- Definición de tipo abstracto de datos
  - Representación de tipos abstractos mediante vectores
  - Representación de tipos abstractos mediante listas
  - Representación de tipos abstractos mediante listas de asociación
7. Lectura y escritura
- Interacción con el sistema
  - Apertura y cierre de un fichero
  - Lectura
  - Escritura
  - Interacción entre funciones y ficheros

## **Segunda parte: Prolog**

8. Introducción al lenguaje Prolog
- Características fundamentales de la programación lógica
  - Hechos
  - Preguntas
  - Variables
  - Conjunción, disyunción y negación
  - Reglas
9. Elementos básicos del Prolog
- Elementos simples y compuestos
  - Aritmética
  - Igualdad y coincidencia
  - Operadores relacionales
10. Listas
- Descripción
  - Operaciones con listas
11. La reevaluación y “el corte”
- Generación de soluciones múltiples

- Descripción del “corte”
- Aplicaciones y problemas del corte

#### 12. Entrada y salida

- Lectura y escritura de términos
- Lectura y escritura de caracteres
- Lectura y escritura con ficheros

### **12. MECANISMOS DE CONTROL Y SEGUIMIENTO**

El profesor podrá comprobar el grado de seguimiento de la asignatura mediante:

- La asistencia a las clases de teoría y prácticas
- La elaboración y exposición del trabajo de teoría.
- La corrección de las prácticas.
- Las tutorías personales.
- Las consultas por correo electrónico.