



UNIVERSIDAD
POLITÉCNICA
DE MADRID

PROCESO DE
COORDINACIÓN DE LAS
ENSEÑANZAS PR/CL/001



E.T.S. de Ingenieros
Informáticos

ANX-PR/CL/001-01

GUÍA DE APRENDIZAJE

ASIGNATURA

105000017 - Programacion Declarativa: Logica Y Restricciones

PLAN DE ESTUDIOS

10II - Grado En Ingenieria Informatica

CURSO ACADÉMICO Y SEMESTRE

2025/26 - Segundo semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos.....	1
2. Profesorado.....	1
3. Conocimientos previos recomendados.....	2
4. Competencias y resultados de aprendizaje.....	2
5. Descripción de la asignatura y temario.....	3
6. Cronograma.....	5
7. Actividades y criterios de evaluación.....	7
8. Recursos didácticos.....	9
9. Otra información.....	10

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	105000017 - Programacion Declarativa: Logica y Restricciones
No de créditos	3 ECTS
Carácter	Obligatoria
Curso	Tercero curso
Semestre	Sexto semestre
Período de impartición	Febrero-Junio
Idioma de impartición	Castellano
Titulación	10II - Grado en Ingenieria Informatica
Centro responsable de la titulación	10 - E.T.S. De Ingenieros Informáticos
Curso académico	2025-26

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
M. Carmen Suarez De Figuroa Baonza	2201	mdelcarmen.suarezdefigueroa@upm.es	Sin horario.
Jose Francisco Morales Caballero	2101	josefrancisco.morales@upm.es	Sin horario.
Miguel Garcia Remesal	2206	miguel.garcia.remesal@upm.es	Sin horario.

Manuel De Hermenegildo Salinas (Coordinador/a)	2212	manuel.hermenegildo@upm. es	Sin horario.
---	------	--------------------------------	--------------

* Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías con el profesorado.

3. Conocimientos previos recomendados

3.1. Asignaturas previas que se recomienda haber cursado

- Algoritmos Y Estructura De Datos
- Lógica
- Programación I
- Programación II

3.2. Otros conocimientos previos recomendados para cursar la asignatura

El plan de estudios Grado en Ingeniería Informática no tiene definidos otros conocimientos previos para esta asignatura.

4. Competencias y resultados de aprendizaje

4.1. Competencias

Ce 2 - Formalización y especificación de problemas reales cuya solución requiere el uso de la informática.

Ce 24 - Elegir y usar los lenguajes de programación adecuados al tipo de aplicación a desarrollar.

Ce 34 - Crear prototipos, simulaciones o modelos que permitan la validación del sistema con el cliente.

Ce 8 - Poseer destrezas fundamentales de la programación que permitan la implementación de los algoritmos y las estructuras de datos en el software.

4.2. Resultados del aprendizaje

RA298 - Modelar declarativamente la solución a un problema y expresarlo elegantemente con un programa lógico eficiente.

RA297 - Conocer los fundamentos de la programación lógica y sus campos de aplicación.

5. Descripción de la asignatura y temario

5.1. Descripción de la asignatura

Esta asignatura presenta la **programación lógica**, uno de los paradigmas fundamentales de programación (junto con la programación funcional, la imperativa, y la orientada a objetos) que se basa en la utilización de la lógica formal como lenguaje práctico para la programación de aplicaciones. La asignatura comienza presentando técnicas de representación y resolución de problemas utilizando *programación lógica pura*. A continuación, se estudia la programación en el *lenguaje Prolog*, así como técnicas de programación eficiente en este lenguaje, con especial énfasis en las aplicaciones en inteligencia artificial. También se presenta una introducción a la **programación lógica con restricciones** y algunos temas avanzados. Es una asignatura eminentemente aplicada, de programación, en la que el alumno realiza diversas prácticas utilizando un sistema avanzado de programación que permite programar con programación lógica pura, Prolog, funciones, orden superior, restricciones, y otras extensiones.

5.2. Temario de la asignatura

1. Introducción

- 1.1. Resolución de problemas y programación declarativa
- 1.2. Qué es (C)LP?

2. Programación lógica pura

- 2.1. Sintaxis y semántica
- 2.2. Resolución y unificación, control de la búsqueda
- 2.3. Estructuras de datos
- 2.4. Programación recursiva
- 2.5. Grafos, autómatas

3. Prolog

- 3.1. Modelo de ejecución
- 3.2. Aritmética
- 3.3. Datos estructurados
- 3.4. Meta-programación, orden superior, predicados de agregación
- 3.5. Inspección y modificación dinámica del programa
- 3.6. Corte y negación
- 3.7. Estructuras de datos incompletas, gramáticas
- 3.8. Técnicas para programación eficiente en Prolog

4. Introducción a la Programación Lógica con Restricciones (CLP)

- 4.1. Satisfacción de restricciones
- 4.2. Técnicas básicas de programación con restricciones (Q/R y dominios finitos)

6. Cronograma

6.1. Cronograma de la asignatura *

Sem	Actividad tipo 1	Actividad tipo 2	Tele-enseñanza	Actividades de evaluación
1	1.1 y 1.2 Duración: 02:00 LM: Actividad del tipo Lección Magistral			Ejercicios de clase (realizados presencialmente durante las distintas clases del curso, puntuables) TI: Técnica del tipo Trabajo Individual Evaluación Progresiva Presencial Duración: 00:00
2	2.1 y 2.2 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
3	2.3 Duración: 01:00 LM: Actividad del tipo Lección Magistral 2.1-2.3 Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio			
4	2.4 Duración: 02:00 LM: Actividad del tipo Lección Magistral			Ejercicios prácticos de programación 1 TI: Técnica del tipo Trabajo Individual Evaluación Progresiva No presencial Duración: 00:00
5	2.5 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
6	2.4-2.5 Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio 3.1 Duración: 01:00 LM: Actividad del tipo Lección Magistral			Ejercicios prácticos de programación 2 TI: Técnica del tipo Trabajo Individual Evaluación Progresiva No presencial Duración: 00:00
7	3.2 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
8	3.3 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
9	3.4 Duración: 01:00 LM: Actividad del tipo Lección Magistral 3.1-3.4 Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio			Práctica de programación 1 TI: Técnica del tipo Trabajo Individual Evaluación Progresiva No presencial Duración: 00:00

10	3.5 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
11	3.6 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
12	3.7 Duración: 02:00 LM: Actividad del tipo Lección Magistral			Práctica de programación 2 TI: Técnica del tipo Trabajo Individual Evaluación Progresiva No presencial Duración: 00:00
13	3.8 Duración: 01:00 LM: Actividad del tipo Lección Magistral 3.5-3.8 Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio			
14	4.1 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
15	4.2 Duración: 01:00 LM: Actividad del tipo Lección Magistral 4.1-4.2 Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio			
16				
17				Evaluación Progresiva: examen práctico ET: Técnica del tipo Prueba Telemática Evaluación Progresiva Presencial Duración: 01:00 Prueba Global: examen ET: Técnica del tipo Prueba Telemática Evaluación Global Presencial Duración: 01:00 Prueba Global: entrega de prácticas ET: Técnica del tipo Prueba Telemática Evaluación Global No presencial Duración: 00:00

Para el cálculo de los valores totales, se estima que por cada crédito ECTS el alumno dedicará dependiendo del plan de estudios, entre 26 y 27 horas de trabajo presencial y no presencial.

7. Actividades y criterios de evaluación

7.1. Actividades de evaluación de la asignatura

7.1.1. Evaluación (progresiva)

Sem.	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
1	Ejercicios de clase (realizados presencialmente durante las distintas clases del curso, puntuables)	TI: Técnica del tipo Trabajo Individual	Presencial	00:00	%	5 / 10	Ce 8 Ce 24
4	Ejercicios prácticos de programación 1	TI: Técnica del tipo Trabajo Individual	No Presencial	00:00	%	5 / 10	Ce 8 Ce 24
6	Ejercicios prácticos de programación 2	TI: Técnica del tipo Trabajo Individual	No Presencial	00:00	%	5 / 10	Ce 8 Ce 24
9	Práctica de programación 1	TI: Técnica del tipo Trabajo Individual	No Presencial	00:00	25%	5 / 10	Ce 8 Ce 24
12	Práctica de programación 2	TI: Técnica del tipo Trabajo Individual	No Presencial	00:00	25%	5 / 10	Ce 8 Ce 24
17	Evaluación Progresiva: examen práctico	ET: Técnica del tipo Prueba Telemática	Presencial	01:00	50%	5 / 10	Ce 2 Ce 8 Ce 24 Ce 34

7.1.2. Prueba evaluación global

Sem	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
17	Prueba Global: examen	ET: Técnica del tipo Prueba Telemática	Presencial	01:00	50%	5 / 10	Ce 2 Ce 8 Ce 24 Ce 34

17	Prueba Global: entrega de prácticas	ET: Técnica del tipo Prueba Telemática	No Presencial	00:00	50%	5 / 10	Ce 2 Ce 8 Ce 24 Ce 34
----	-------------------------------------	--	---------------	-------	-----	--------	--------------------------------

7.1.3. Evaluación convocatoria extraordinaria

Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
Extraordinaria: examen	ET: Técnica del tipo Prueba Telemática	Presencial	01:00	50%	5 / 10	
Extraordinaria: entrega prácticas	ET: Técnica del tipo Prueba Telemática	Presencial	00:00	50%	5 / 10	

7.2. Criterios de evaluación

Sistema de evaluación de la asignatura:

El sistema normal de evaluación de la asignatura, que se aplica a todos los alumnos, es por *evaluación progresiva*.

- Al ser una asignatura eminentemente práctica, además de diversos ejercicios de clase y 2 conjuntos de ejercicios prácticos, se realizarán 2 prácticas evaluables durante el curso.
- Dichas prácticas se realizarán de forma individual.
- Los enunciados, instrucciones, información sobre los sistemas a utilizar, etc., se enviarán por correo electrónico y Moodle.
- La nota final de las prácticas será la media de las dos prácticas y se tendrán en cuenta los ejercicios, incluidos los de clase. No hay nota mínima para cada práctica y es posible realizar una práctica o ejercicio aunque no se haya entregado o aprobado otra.
- Además de las prácticas habrá un examen práctico individual, que se realizará en línea.
- La nota final será la media entre las prácticas y el examen.
- Por tanto, las prácticas valen un 50% de la nota final y el examen un 50%.

Para aprobar la asignatura **en cualquier modalidad** hay que aprobar tanto las prácticas (nota mínima ≥ 5.0) como el examen individual (nota mínima ≥ 5.0)

Evaluación mediante prueba global:

La evaluación mediante prueba global consistirá también en la realización de un examen práctico individual. Además, como recuperación de las prácticas, se podrá entregar el mismo día de la prueba global la práctica o prácticas que no se hubieran entregado o se hubieran suspendido anteriormente, en las que se podrá exigir apartados distintos o adicionales. La puntuación de dichas prácticas hará media con la del examen, pero no servirá en este caso para obtener una nota superior a la del examen.

Evaluación extraordinaria:

Aquellos estudiantes que no hayan superado la asignatura podrán presentarse a la evaluación extraordinaria. Ésta consistirá también en la realización de un examen práctico individual, y también se podrán entregar el mismo día la práctica o prácticas que no se hubieran entregado o se hubieran suspendido anteriormente, en las que se podrá exigir apartados distintos o adicionales. La puntuación de dichas prácticas hará media con la del examen, pero no servirá en este caso para obtener una nota superior a la del examen.

8. Recursos didácticos

8.1. Recursos didácticos de la asignatura

Nombre	Tipo	Observaciones
``The Art of Prolog" (Second edition), Sterling & Shapiro, MIT Press, 1994.	Bibliografía	
``From Logic Programming to Prolog", K. Apt, Prentice-Hall, 1997.	Bibliografía	
``Prolog Programming for Artificial Intelligence", I. Bratko, Addison-Wesley Ltd. 1990 (2nd edition); 2000 (3rd edition).	Bibliografía	
``Programming in Prolog", Clocksin & Mellish, 1981, Springer-Verlag.	Bibliografía	
``Programming with Constraints: An Introduction", Marriott & Stuckey, MIT Press, 1998.	Bibliografía	

``Essentials of Logic Programming'', C. Hogger, 1990, Clarendon Press, Oxford.	Bibliografía	
Sitio Moodle de la asignatura	Recursos web	
Página web de la asignatura	Recursos web	http://www.clip.dia.fi.upm.es/prode
Sistema de programación Ciao Prolog	Equipamiento	https://ciao-lang.org
Patio de juegos del sistema Ciao Prolog	Equipamiento	https://ciao-lang.org/playground
Entorno de programación para Ciao en Emacs	Equipamiento	https://ciao-lang.org/install.html
Entorno de programación para Ciao en Visual Studio Code	Equipamiento	https://marketplace.visualstudio.com/items?itemName=ciao-lang.ciao-prolog-vscode
Aula	Equipamiento	
Laboratorio: Centro de Cálculo	Equipamiento	

9. Otra información

9.1. Otra información sobre la asignatura

- Esta asignatura está relacionada con el "**Objetivo de Desarrollo Sostenible 9**" (Industria, innovación e infraestructura) definido por el Programa de las Naciones Unidas para el Desarrollo (www.undp.org) en lo referente a innovación e investigación científica en tecnologías de la información.

Innovación docente:

Las siguientes acciones "learning by doing":

- Se proporciona a los alumnos entornos de programación desarrollados para la asignatura en diversos IDEs

(Emacs, VSC, etc.) así como un playground en el navegador que no necesita instalación.

- Se proporcionan enlaces desde las transparencias a ejercicios y ejemplos interactivos online con auto-corrección en la plataforma (playgorund) desarrollada ex-profeso para la signatura. Esto permite realizar asignar y realizar ejercicios tanto durante las clases como en casa.
- Además de las transparencias y textos se proporcionan algunos vídeos de ayuda para la instalación y uso de los sistemas de programación utilizados y para algunos temas concretos de la asignatura.
- Para los ejercicios puntuables se utilizan auto-correctores sobre la plataforma Deliverit, los cuales, al proporcionar realimentación directa a los alumnos sobre sus prácticas, son elementos didácticos muy importantes.