



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

105000439 - El Arte De Programar

PLAN DE ESTUDIOS

10II - Grado En Ingenieria Informatica

CURSO ACADÉMICO Y SEMESTRE

2024/25 - Primer semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos.....	1
2. Profesorado.....	1
3. Conocimientos previos recomendados.....	3
4. Competencias y resultados de aprendizaje.....	3
5. Descripción de la asignatura y temario.....	4
6. Cronograma.....	7
7. Actividades y criterios de evaluación.....	10
8. Recursos didácticos.....	15
9. Otra información.....	16
10. Adendas.....	17

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	105000439 - El Arte de Programar
No de créditos	6 ECTS
Carácter	Optativa
Curso	Tercero curso
Semestre	Quinto semestre
Período de impartición	Septiembre-Enero
Idioma de impartición	Castellano
Titulación	10II - Grado en Ingeniería Informática
Centro responsable de la titulación	10 - E.T.S. De Ingenieros Informáticos
Curso académico	2024-25

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
Lars-ake Fredlund	D2309	larsake.fredlund@upm.es	L - 10:00 - 13:00 J - 10:00 - 13:00 Se recomienda solicitar cita por correo electrónico al menos con 24 horas de antelación

Julio Mariño Carballo	D2308	julio.marino@upm.es	M - 15:00 - 17:00 X - 10:00 - 12:00 X - 15:00 - 17:00 Se recomienda solicitar cita por correo electrónico al menos con 24 horas de antelación
Clara Benac Earle	D2302	clara.benac@upm.es	M - 10:00 - 12:00 J - 12:00 - 14:00 V - 12:00 - 14:00 Se recomienda solicitar cita por correo electrónico al menos con 24 horas de antelación
Santiago Tapia Fernandez (Coordinador/a)	D2307	santiago.tapia@upm.es	M - 10:00 - 13:00 X - 10:00 - 13:00 Los alumnos deben contactar para las tutorías a través de TEAMS.

* 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

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

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

CG-1/21 - Capacidad de resolución de problemas aplicando conocimientos de matemáticas, ciencias e ingeniería.

CG-2/CE45 - Capacidad para el aprendizaje autónomo y la actualización de conocimientos, y reconocimiento de su necesidad en el área de la informática.

CG-3/4 - Saber trabajar en situaciones carentes de información y bajo presión, teniendo nuevas ideas, siendo creativo.

CG-6 - Capacidad de abstracción, análisis y síntesis

CG-7:10/16/17 - Capacidad para trabajar dentro de un equipo, organizando, planificando, tomando decisiones, negociando y resolviendo conflictos, relacionándose, y criticando y haciendo autocrítica

Ce 19/20 - Conocimiento de los tipos apropiados de soluciones, y comprensión de la complejidad de los problemas informáticos y la viabilidad de su solución.

4.2. Resultados del aprendizaje

RA278 - Desarrollar la solución matemática y algorítmica mas apropiada a un problema informático que requiera un tratamiento especialmente complejo, analizando y exponiendo su viabilidad.

5. Descripción de la asignatura y temario

5.1. Descripción de la asignatura

En esta asignatura se estudiarán los algoritmos en más detalle que en las asignaturas obligatorias del grado. Prestaremos especial atención a cómo comparar diferentes algoritmos, midiendo su tiempo de ejecución o consumo de memoria. Asimismo introduciremos nociones abstractas de complejidad y tratabilidad con las que aprenderemos a reconocer problemas especialmente complicados de tratar algorítmicamente. Una gran parte de la asignatura consistirá en presentar técnicas útiles para el diseño de algoritmos eficientes.

Uno de los cometidos de esta asignatura es animaros y prepararos para participar en concursos de programación, tales como el ACM ICPC (<https://icpc.global>) o el Ada Byron (<https://ada-byron.es>). En concreto, usaremos algunos problemas de ediciones recientes como retos o para ilustrar algunas de las técnicas y conceptos presentados.

La asignatura se compone de:

1. Una parte teórica, impartida mediante clases tipo lección magistral donde se explicarán complejidad, tratabilidad y distintas técnicas algorítmicas. Algunos ejemplos de estas técnicas son: divide y vencerás, programación dinámica, búsquedas heurísticas, etc.
2. Una parte práctica, impartida mediante clases prácticas con ordenador. Estas prácticas se van a orientar a la resolución de problemas tipo concurso de programación, en cada una de las clases se propondrán 2 ó 3 problemas a realizar en equipo entre los alumnos presentes. Se orientará a los alumnos tanto en general, por ejemplo: formas de trabajar con la entrada y salida de datos, depuración de errores, realización de pruebas, etc. como en particular sobre aspectos concretos de cada problema. Según avance el curso para la resolución de algunos problemas será necesario utilizar las técnicas algorítmicas vistas en la parte

teórica.

El enfoque orientado a la resolución de problemas permite mejorar significativamente las habilidades relacionadas con el desarrollo de programas, desde el análisis y planteamiento del problema hasta la implementación y la eventual depuración de errores. Así mismo, permite aplicar los conocimientos teóricos, la resolución de algunos problemas se basa en la aplicación de los algoritmos vistos en la parte teórica. En las clases prácticas los equipos serán aleatorios, en ningún caso los alumnos pondrán escoger a sus compañeros. Esta forma de asignación de equipos permite mejorar competencias de trabajo en equipo y, en cierta manera, aumentar las habilidades relacionadas con el desarrollo de programas porque los alumnos también aprenden unos de otros.

No hay un lenguaje de programación concreto como requisito en la asignatura. Aunque los aspectos teóricos y prácticos, en general, se explicarán en Java, los alumnos podrán elegir hacer los problemas en: Java, C++, C o Python. En este sentido, la asignatura puede ser una buena ocasión para aprender otro lenguaje o profundizar en cualquiera de ellos.

El temario propuesto **no** es un calendario. La planificación de los temas se ajusta cada curso atendiendo a criterios académicos.

5.2. Temario de la asignatura

1. Introducción

- 1.1. Presentación de la asignatura. Programación competitiva.
- 1.2. Familiarización con el entorno de trabajo. Jueces.

2. Eficiencia y complejidad de algoritmos

- 2.1. Análisis de tiempo de ejecución de algoritmos. Otras medidas de eficiencia.
- 2.2. Complejidad de algoritmos. Notación $O()$.
- 2.3. Tratabilidad. Clases de complejidad. Reducciones.

3. Técnicas básicas

- 3.1. Tratamiento de cadenas alfanuméricas (Strings).
- 3.2. Estructuras de datos.
- 3.3. Algoritmos sobre árboles y otros grafos.

4. Técnicas de diseño de algoritmos

- 4.1. Divide y vencerás
- 4.2. Algoritmos voraces
- 4.3. Búsqueda combinatoria. Backtracking.
- 4.4. Búsqueda heurística
- 4.5. Programación dinámica

6. Cronograma

6.1. Cronograma de la asignatura *

Sem	Actividad tipo 1	Actividad tipo 2	Tele-enseñanza	Actividades de evaluación
1	Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral			
2	Clase de Resolución de Problemas Duración: 02:00 G: Gamificación Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral			
3	Clase de Resolución de Problemas Duración: 02:00 G: Gamificación Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral			
4	Clase de Resolución de Problemas Duración: 02:00 G: Gamificación Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral			
5	Clase de Resolución de Problemas Duración: 02:00 G: Gamificación Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral			
6	Clase de Resolución de Problemas Duración: 02:00 G: Gamificación Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral			

7	<p>Clase de Resolución de Problemas Duración: 02:00 G: Gamificación</p> <p>Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			
8	<p>Clase de Resolución de Problemas Duración: 02:00 G: Gamificación</p> <p>Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			<p>Concursos de Programación TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva Presencial Duración: 00:00</p>
9	<p>Clase de Resolución de Problemas Duración: 02:00 G: Gamificación</p> <p>Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			<p>Concursos de Programación TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva Presencial Duración: 00:00</p>
10	<p>Clase de Resolución de Problemas Duración: 02:00 G: Gamificación</p> <p>Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			<p>Concursos de Programación TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva Presencial Duración: 00:00</p>
11	<p>Clase de Resolución de Problemas Duración: 02:00 G: Gamificación</p> <p>Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>	<p>Examen Práctico de Evaluación Progresiva Duración: 04:00 OT: Otras actividades formativas / Evaluación</p>		<p>Concursos de Programación TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva Presencial Duración: 00:00</p> <p>Fase Local Ada Byron OT: Otras técnicas evaluativas Evaluación Progresiva Presencial Duración: 04:00</p>
12	<p>Clase de Resolución de Problemas Duración: 02:00 G: Gamificación</p> <p>Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			<p>Concursos de Programación TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva Presencial Duración: 00:00</p>
13	<p>Clase de Resolución de Problemas Duración: 02:00 G: Gamificación</p> <p>Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			<p>Concursos de Programación TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva Presencial Duración: 00:00</p>
14	<p>Clase de Resolución de Problemas Duración: 02:00 G: Gamificación</p> <p>Clase de Teoría Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			<p>Concursos de Programación TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva Presencial Duración: 00:00</p>

15	Evaluación Duración: 04:00 OT: Otras actividades formativas / Evaluación			Concursos de Programación TG: Técnica del tipo Trabajo en Grupo Evaluación Progresiva Presencial Duración: 03:30 Test teoría EX: Técnica del tipo Examen Escrito Evaluación Progresiva Presencial Duración: 00:30
16				
17				Recuperación concursos EP: Técnica del tipo Examen de Prácticas Evaluación Global Presencial Duración: 03:30 Test teoría EX: Técnica del tipo Examen Escrito Evaluación Global Presencial Duración: 00:30

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
8	Concursos de Programación	TG: Técnica del tipo Trabajo en Grupo	Presencial	00:00	7%	0 / 10	CG-3/4 CG-6 CG-7:10/16/17 CG-1/21 CG-2/CE45 Ce 19/20
9	Concursos de Programación	TG: Técnica del tipo Trabajo en Grupo	Presencial	00:00	7%	0 / 10	CG-3/4 CG-6 CG-7:10/16/17 CG-1/21 CG-2/CE45 Ce 19/20
10	Concursos de Programación	TG: Técnica del tipo Trabajo en Grupo	Presencial	00:00	7%	0 / 10	CG-3/4 CG-6 CG-7:10/16/17 CG-1/21 CG-2/CE45 Ce 19/20
11	Concursos de Programación	TG: Técnica del tipo Trabajo en Grupo	Presencial	00:00	7%	0 / 10	CG-3/4 CG-6 CG-7:10/16/17 CG-1/21 CG-2/CE45 Ce 19/20
11	Fase Local Ada Byron	OT: Otras técnicas evaluativas	Presencial	04:00	10%	0 / 10	CG-1/21 CG-2/CE45 CG-3/4 CG-6 CG-7:10/16/17 Ce 19/20
12	Concursos de Programación	TG: Técnica del tipo Trabajo en Grupo	Presencial	00:00	8%	0 / 10	CG-1/21 CG-2/CE45 CG-3/4 CG-6 CG-7:10/16/17 Ce 19/20

13	Concursos de Programación	TG: Técnica del tipo Trabajo en Grupo	Presencial	00:00	8%	0 / 10	CG-1/21 CG-2/CE45 CG-3/4 CG-6 CG-7:10/16/17 Ce 19/20
14	Concursos de Programación	TG: Técnica del tipo Trabajo en Grupo	Presencial	00:00	8%	0 / 10	CG-1/21 CG-2/CE45 CG-3/4 CG-6 CG-7:10/16/17 Ce 19/20
15	Concursos de Programación	TG: Técnica del tipo Trabajo en Grupo	Presencial	03:30	8%	0 / 10	
15	Test teoría	EX: Técnica del tipo Examen Escrito	Presencial	00:30	30%	5 / 10	

7.1.2. Prueba evaluación global

Sem	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
17	Recuperación concursos	EP: Técnica del tipo Examen de Prácticas	Presencial	03:30	70%	5 / 10	CG-1/21 CG-2/CE45 CG-3/4 CG-6 CG-7:10/16/17 Ce 19/20
17	Test teoría	EX: Técnica del tipo Examen Escrito	Presencial	00:30	30%	4 / 10	CG-2/CE45 CG-3/4 CG-6 Ce 19/20

7.1.3. Evaluación convocatoria extraordinaria

Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
Examen Práctico	TG: Técnica del tipo Trabajo en Grupo	Presencial	03:00	70%	5 / 10	CG-6 CG-3/4 CG-2/CE45 CG-1/21 Ce 19/20 CG-7:10/16/17

Test Teoría	EX: Técnica del tipo Examen Escrito	Presencial	00:30	30%	4 / 10	CG-6 CG-3/4 CG-2/CE45 Ce 19/20
-------------	--	------------	-------	-----	--------	---

7.2. Criterios de evaluación

Descripción general

Todas las evaluaciones indicadas en esta guía y que se corresponden con la descripción "**Concursos de Programación**" van a consistir en **resolver problemas** prácticos en **horario de clase, presencialmente** y en **equipo**. No se pueden hacer en remoto ni en otro horario. El trabajo en **equipo** consiste en la **participación activa** en la resolución de los problemas. **Los alumnos que no participen serán excluidos de su equipo de forma inmediata.**

Adicionalmente, en la guía aparece como actividad de evaluación aparte la "**Fase Local Ada Byron**", tal y como se indica esta fase local del concurso entre los alumnos de la UPM será una actividad docente de la propia asignatura donde podrán participar otros alumnos sin matricular en la asignatura, pero que será también, a todos los efectos, una actividad regular de la propia asignatura tal y como lo serán los concursos semanales.

Los **concursos no se pueden repetir**, pero **sí recuperar**. La **recuperación** de los *concursos* será **en la "Recuperación Concursos"** de la evaluación global. Se recuperarán sólo aquellos concursos que no se hayan realizado en su momento, los que sí se hayan realizado y superado se conservan para la convocatoria ordinaria (global), pero no para la extraordinaria.

Un concurso consiste en el desarrollo de programas de ordenador que resuelven un enunciado y que se someten a pruebas automáticas en un juez en línea. En la asignatura se podrá utilizar cualquier juez en línea, pero lo más probable es que se utilice el juez de **acceptaelreto.com**. Se entiende que un **problema está resuelto** cuando se obtiene el correspondiente veredicto "**aceptado**" en el juez.

Dado que la evaluación de la parte práctica de la asignatura se basa en estos concursos, la asignación de una nota a cada una de las pruebas de evaluación y el cálculo posterior de la nota a partir de una media ponderada no resulta apropiada. Por eso la evaluación se establecerá en base a un sistema de puntos tal y como se indica más adelante.

Evaluación Progresiva

La nota de evaluación progresiva se calcula como $0,3 * (\text{Test de Teoría}) + 0,7 * (\text{Nota Concursos})$.

El Test de Teoría es una prueba escrita individual que consiste en un examen con preguntas tipo test o de respuesta corta. El Test de Teoría es recuperable en evaluación global.

Para calcular la "**Nota Concursos**", se suman los puntos obtenidos de la siguiente forma:

1. Se obtendrán **0.5 puntos** por problema resuelto en cualquiera de los "Concursos de Programación" que figuran como pruebas de evaluación de la asignatura. Hasta un total de 10 problemas, es decir, **un máximo de 5 puntos**.
2. Se obtendrán puntos extra por la **rapidez** en la resolución del problema o por la **eficiencia** de la solución. Hasta un **máximo de 2,5 puntos**.
3. Se obtendrá hasta un máximo de **1 punto** según el puesto alcanzado en la clasificación en la **fase local del Ada Byron**.
4. Solamente para aquellos alumnos que hayan alcanzado una puntuación **mínima de 7 puntos** en los ítems anteriores se obtendrán **0,5 puntos** por la resolución de problemas adicionales resueltos **individualmente** hasta **un máximo de 2,5 puntos**.

Evaluación Global

La nota de evaluación global se calcula como $0,3 * (\text{Test de Teoría}) + 0,7 * (\text{Nota de Concursos})$. Los alumnos sólo tendrán que presentar a las pruebas de evaluación que hayan suspendido previamente en la evaluación progresiva. Es decir, se conserva la nota del "Test de Teoría" o la "Nota de Concursos" cuando sea mayor o igual que 5.

El Test de Teoría tiene el mismo formato que el de evaluación progresiva.

La "Recuperación concursos" se compone de problemas similares a los propuestos en la evaluación progresiva y

permitirá recuperar la puntuación de los concursos de la evaluación progresiva. Sólo será necesario recuperar los concursos no superados, no es necesario recuperar aquellos problemas ya resueltos durante la evaluación progresiva. Esta prueba se realiza **individualmente**.

Nota de Actas

La nota final (en actas) será la nota mejor entre la nota por evaluación progresiva (NP) y la nota por evaluación global (NG). Para aprobar la asignatura la nota final tiene que ser de un mínimo de 5.

Convocatoria extraordinaria

La forma de evaluación de la convocatoria extraordinaria se compone de dos pruebas:

- El Test Teoría es análogo a los de la convocatoria ordinaria.
- El Examen Práctico son una serie de problemas a resolver por el alumno equivalentes a aquellos que se han propuesto en los concursos de la convocatoria ordinaria. Para la calificación de estos problemas se aplica el mismo criterio que en los concursos, es necesario que el juez en línea de la solución como "Aceptada".

Advertencia sobre Fraude Académico.

Los alumnos que cometan cualquier tipo de fraude académico durante el desarrollo de los concursos o en el test de teoría tendrán suspensa la convocatoria ordinaria.

Se entiende por fraude académico la entrega de soluciones que no hayan sido íntegramente escritas por los alumnos del equipo. Se **incluye explícitamente como fraude el uso de Inteligencia Artificial (Chat GPT, copilot, etc.)**. En los concursos realizados en equipo el fraude afecta a todos los miembros del equipo.

8. Recursos didácticos

8.1. Recursos didácticos de la asignatura

Nombre	Tipo	Observaciones
The Algorithm Design Manual	Bibliografía	Steven S. Skiena. The Algorithm Design Manual. Springer, London, 2008.
Programming Challenges: The Programming Contest Training Manual	Bibliografía	Steven S. Skiena and Miguel Revilla. Programming Challenges: The Programming Contest Training Manual. Springer-Verlag, Berlin, Heidelberg, 2003.
The Art of Computer Programming, Volume 3: (2nd Ed.) Sorting and Searching.	Bibliografía	Donald E. Knuth. The Art of Computer Programming, Volume 3: (2nd Ed.) Sorting and Searching. Addison Wesley Longman Publishing Co., Inc., USA, 1998.
The Art of Computer Programming: Combinatorial Algorithms, Part 1.	Bibliografía	Donald E. Knuth. The Art of Computer Programming: Combinatorial Algorithms, Part 1. Addison-Wesley Professional, 2011.
Guide to Competitive Programming: Learning and Improving Algorithms Through Contests.	Bibliografía	Antti Laaksonen. Guide to Competitive Programming: Learning and Improving Algorithms Through Contests. SDpringer Verlag, 2017.

9. Otra información

9.1. Otra información sobre la asignatura

La asignatura se relaciona con el ODS4.

10. Adendas

- Se modifican los siguientes puntos de la Guía: * Cronograma. Se añaden dos nuevos concursos iguales a los ya programados en las semanas 6 y 7. De esta forma los alumnos tendrán un total de 8 concursos. * En evaluación progresiva. Se añaden los nuevos concursos a efectos de calcular los puntos por concursos. Y, además, se añade un ítem más en la evaluación de los mismos, detrás del número 4: "Aquellos alumnos que hayan obtenido los 5 puntos en concursos previos podrán optar a resolver problemas más difíciles en los concursos habituales. Estos problemas sí contabilizarán para notas por encima del 5."