



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

GUÍA DOCENTE DE COMPLEJIDAD Y COMPUTACIÓN

La presente guía docente corresponde a la asignatura Complejidad y Computación (CyC), aprobada para el curso lectivo 2017-2018 en Junta de Centro y publicada en su versión definitiva en la página web de la Escuela Politécnica Superior. La guía docente de CyC aprobada y publicada antes del periodo de matrícula tiene el carácter de contrato con el estudiante.



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

ASIGNATURA

COMPLEJIDAD Y COMPUTACIÓN (CyC)

1.1. Código

18775 del Grado en Ingeniería Informática

1.2. Materia

Computación e Inteligencia Artificial

1.3. Tipo

Optativa

1.4. Nivel

Grado

1.5. Curso

4º

1.6. Semestre

2º

1.7. Número de créditos

6 créditos ECTS

1.8. Requisitos previos

Ninguno

1.9. Requisitos mínimos de asistencia a las sesiones presenciales

Se plantean dos itinerarios, uno con asistencia obligatoria a clase y otro sin ella, los estudiantes deberán optar por uno u otro desde el principio del curso y cumplir con



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

los distintos requisitos de evaluación que conlleva cada uno de los modelos, publicados en la presente guía docente (ver apartado 4).

ITINERARIO CON ASISTENCIA OBLIGATORIA A CLASE

La asistencia es obligatoria al menos en un 85%.

ITINERARIO SIN ASISTENCIA OBLIGATORIA A CLASE

La asistencia es muy recomendable aunque no obligatoria.

1.10. Datos del equipo docente

Nota: se debe añadir @uam.es a todas las direcciones de correo electrónico.

Profesor/a coordinador/a de la asignatura:

Dr. Xavier Alamán Roldán

Departamento de Ingeniería Informática

Escuela Politécnica Superior

Despacho - Módulo: B-420 Edificio B - 4ª Planta

Teléfono: +34 91 497 2250

Correo electrónico: xavier.alaman

Página web: <http://arantxa.ii.uam.es/~xalaman>

Horario de atención al alumnado: Petición de cita previa por correo electrónico.

La información relativa a los demás profesores del equipo docente está en:
<http://www.uam.es/ss/Satellite/EscuelaPolitecnica/es/estudios/grado-3/Page/contenidoFinal/grado-en-ingenieria-informatica-3.htm>.



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

1.11. Objetivos del curso

El curso proporciona una introducción a las ideas centrales de la teoría de la computación. Partiendo del conocimiento del estudiante sobre autómatas finitos y sobre máquinas de Turing -adquiridos en asignaturas previas del grado en Ingeniería Informática-, se exploran los límites de lo computable (qué problemas pueden o no resolverse por distintos tipos de máquinas) y del razonamiento (teorema de Gödel). Finalmente se estudia el concepto de complejidad de los problemas computacionales (problemas NP y NP completos). También se introducen modelos alternativos de computación (i.e. computación cuántica) y se realiza un análisis de sus potenciales aplicaciones y limitaciones.

Las **competencias** que se pretenden adquirir con esta asignatura son:

De tecnología específica:

CC5: Capacidad para adquirir, obtener, formalizar y representar el conocimiento humano en una forma computable para la resolución de problemas mediante un sistema informático en cualquier ámbito de aplicación, particularmente los relacionados con aspectos de computación, percepción y actuación en entornos inteligentes.

CC1: Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar, y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática.

CC2: Capacidad para conocer los fundamentos teóricos de los lenguajes de programación y las técnicas de procesamiento léxico, sintáctico y semántico asociadas, y saber aplicarlas para la creación, diseño y procesamiento de lenguajes.

CC3: Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.

Los objetivos que se pretenden alcanzar con esta asignatura son:

OBJETIVOS GENERALES	
G1	Entender el concepto de “computabilidad” y sus limitaciones.
G2	Entender cómo los sistemas formales pueden encarnar el concepto de “verdadero”, así como la limitación que impone el Teorema de Gödel y sus consecuencias.
G3	Entender el concepto de computacionalmente “tratable”, y saber caracterizar las clases de problemas P, NP y NP-completo.
G4	Entender cómo la computación cuántica permite ir más allá de la máquina de Turing en relación con los problemas intratables.



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

OBJETIVOS ESPECIFICOS POR UNIDAD	
UNIDAD 1.- ¿Qué es computar? Problemas no computables, el teorema de Turing.	
1.1.	Entender cómo los sistemas formales encarnan el concepto de “computable”.
1.2.	Entender el Cálculo Lambda y cómo puede ser empleado para definir “computabilidad”.
1.3.	Entender la propuesta de Turing para la definición de “computable”: la máquina de Turing.
1.4.	Conocer la tesis de Church-Turing y sus implicaciones.
1.5.	Ser capaz de demostrar el teorema de la parada de Turing.
1.6.	Conocer otros problemas no computables.
UNIDAD 2.- ¿Qué es verdadero? El teorema de Gödel.	
2.1.	Entender cómo los sistemas formales encarnan el concepto de “verdadero”.
2.2.	Ser capaz de identificar o definir procedimientos de decisión.
2.3.	Conocer las relaciones entre forma y significado.
2.4.	Ser capaz de formalizar el concepto de “verdadero” empleando lógica formal.
2.5.	Entender la teoría de los números naturales y cómo esta teoría engloba a cualquier otro sistema formal.
2.6.	Ser capaz de demostrar el teorema de Gödel.
UNIDAD 3.- ¿Qué es tratable computacionalmente?	
3.1.	Conocer el concepto de Máquina de Turing no determinista.
3.2.	Entender la caracterización de un problema como P o NP.
3.3.	Conocer la definición del problema de la satisfacibilidad y saber aplicar el algoritmo de Davis-Putnam.
3.4.	Entender la caracterización de un problema como NP-Completo y NP-Duro
3.5.	Ser capaz de demostrar el algoritmo de Cook.
3.6.	Ser capaz de aplicar algoritmos aproximados para algunos problemas NP-Completo.
UNIDAD 4.- Más allá de Turing: computación cuántica	
4.1.	Entender el concepto de Qbit.
4.2.	Conocer los fundamentos de la computación cuántica: entrelazamiento de Qbits, puertas cuánticas, el problema de la lectura.
4.3.	Ser capaz de formular algoritmos sencillos usando puertas cuánticas.
4.4.	Entender el algoritmo de Shor y cómo compromete los sistemas de criptografía clásicos.
4.5.	Entender cómo funciona la criptografía cuántica.
4.6.	Entender la teleportación cuántica y su relación con la transmisión de información cuántica.



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

1.12. Contenidos del programa

Programa Sintético

UNIDAD 1. ¿Qué es computar? Problemas no computables, el teorema de Turing.
UNIDAD 2. ¿Qué es verdadero? El teorema de Gödel.
UNIDAD 3. ¿Qué es computacionalmente tratable? El teorema de Cook.
UNIDAD 4. Más allá de Turing: computación cuántica.

Programa Detallado

1. **¿Qué es computar?**
 - 1.1. El concepto de “computable” según Post: sistemas formales.
 - 1.2. El concepto de “computable” según Church: cálculo Lambda.
 - 1.3. El concepto de “computable” según Turing: Máquinas de Turing.
 - 1.4. La tesis de Church/Turing.
 - 1.5. El teorema de la parada de Turing.
 - 1.6. Otros problemas no computables.
2. **¿Qué es verdadero?**
 - 2.1. Sistemas formales. Ejemplos.
 - 2.2. Procedimientos de decisión.
 - 2.3. Forma y significado.
 - 2.4. Formalizando el concepto de “verdadero”: lógica formal.
 - 2.5. Ejemplo: formalización de la teoría de los números naturales.
 - 2.6. El teorema de Gödel .
3. **¿Qué es tratable computacionalmente?**
 - 3.1. Máquinas de Turing no deterministas.
 - 3.2. Algoritmos P y NP.
 - 3.3. El problema de la satisfacibilidad.
 - 3.4. NP-Compleitud: el teorema de Cook
 - 3.5. Algoritmos aproximados para problemas NP-completos.
4. **Más allá de Turing: computación cuántica**
 - 4.1. Bits y Qbits.
 - 4.2. Computación cuántica: fundamentos.
 - 4.3. El algoritmo de Shor para la factorización en números primos.
 - 4.4. Criptografía cuántica.
 - 4.5. Teleportación cuántica.



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

1.13. Referencias de consulta

Bibliografía:

1. “Gödel, Escher, Bach...”, D.R. Hofstadter.
2. “The Turing Omnibus”, A.K. Dewdney.
3. “Explorations in Quantum Computing”, C.P. Williams y S.H. Clearwater.
4. “Feynman Lectures on Computation”, R.P. Feynman.
5. “Computational Complexity”, C.H. Papadimitriou.

Material electrónico de trabajo: los documentos electrónicos de trabajo, se publican en la plataforma Moodle (<http://uam-virtual.es>)

2. Métodos docentes

La metodología utilizada en el desarrollo de la actividad docente incluye los siguientes tipos de actividades:

La organización temporal del curso incluye 3 horas semanales de clases teóricas y 2 horas semanales de clases prácticas/ejercicios.

*Clases de teoría:

Actividad del profesor

Clases expositivas que constarán de:

- Una introducción a los objetivos a alcanzar definidos para cada Tema y Unidad.
- Presentación detallada del Tema/Unidad.
- Se utilizará la pizarra y la presentación mediante transparencias.

Actividad del estudiante:

Actividad presencial: Presentación de temas asignados, toma de apuntes, participación activa en clase respondiendo a las cuestiones planteadas, tanto de manera oral como por escrito. Lectura, resumen y comentario de documentos durante el desarrollo de las clases.

Actividad no presencial: lectura del material bibliográfico y de apoyo, estudio de la materia y realización de los trabajos planteados.

*Clases de practicas:

Actividad del profesor

- Introducción a los objetivos del trabajo a desarrollar por los estudiantes (en grupo o individual).
- Dinamización de la actividad desarrollada en la clase y de la participación de los estudiantes



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

- Orientar y Tutorizar sobre la realización del trabajo y los proyectos.

Actividad del estudiante:

Actividad presencial:

- Resolución de ejercicios y cuestiones teórico-prácticas.
- Lectura de documentos.
- Discusión en grupo.
- Presentación de trabajos en público.
- Programación de proyectos asignados.

Actividad no presencial:

- Resolución de ejercicios.
- Búsqueda de información en Bibliotecas e Internet.
- Lectura de documentos.
- Redacción de documentos y proyectos.
- Elaboración de presentaciones.
- Trabajo en equipo.
- Programación de proyectos asignados.

3. Tiempo de trabajo del estudiante

		Nº de horas	Porcentaje
Presencial	Clases teóricas	42 (60%)	46%
	Clases prácticas	28 (40%)	
No presencial	Búsqueda y Lectura de Documentación	10 (12,5%)	53%
	Elaboración de ejercicios	10 (12,5%)	
	Preparación de presentaciones o trabajos	25 (31%)	
	Programación de proyectos	25 (31%)	
	Otras actividades	10 (12,5%)	
Carga total de horas de trabajo: 25 horas x 6 ECTS		150 h	



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

4. Métodos de evaluación y porcentaje en la calificación final

Método de evaluación continúa

La nota final de la asignatura se obtiene a partir de las notas obtenidas en tres actividades que el alumno tiene que realizar (cada una puntúa por un tercio de la nota). Las actividades serán asignadas, según los intereses del alumno y según las circunstancias del curso, entre las siguientes:

- Comentarios sobre lecturas, ejercicios y mini-exámenes de 5 minutos.
- Elaboración de trabajos escritos relativos al temario de la asignatura.
- Presentaciones en clase de temas asignados por el profesor.
- Proyectos de programación de algoritmos estudiados en clase.

Además tendrán que haber participado con aprovechamiento en las actividades del curso, y realizado los ejercicios y otras actividades diarias que se encomienden.

Cada semana los estudiantes tendrán que realizar una lectura obligatoria, que luego será comentada en clase. Antes de comentar la lectura habrá un mini-examen escrito de 5 minutos para comprobar que todos los estudiantes han realizado la lectura. Aquellos estudiantes que no aprueben este examen tendrán una segunda oportunidad en la siguiente clase, mediante un examen de 15 minutos. Si tampoco lo aprueban, no liberarán la materia correspondiente, y tendrán que presentarse a un examen final (sólo de la materia no aprobada).

En las clases en las que otros alumnos presentan un tema también podrá haber un mini-examen de 5 minutos al final de la clase para comprobar que todos los estudiantes han atendido y han comprendido el tema expuesto. Aquellos que no lo aprueben tendrán que pasar un examen de 15 minutos sobre la materia, y en caso de no aprobarlo tendrán que presentarse a un examen final sobre dicha materia.

Para aprobar la asignatura es obligatorio obtener una nota final mayor o igual a 5 puntos en cada una de las tres actividades realizadas.

Método de evaluación no continúa.

- Tendrán que hacer un examen final sobre el temario de la asignatura.

Convocatoria extraordinaria.

- Tendrán que hacer un examen final sobre el temario de la asignatura.



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

5. Cronograma

Semana	Actividad Presencial	Horas no presenciales
1	- Presentación y motivación de la asignatura, descripción del programa, normativa y los métodos de evaluación, descripción de la plataforma Moodle. - Unidad 1 ¿Qué es computar? Tema 1.1	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación.
2	- Unidad 1 ¿Qué es computar? Tema 1.2	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
3	- Unidad 1 ¿Qué es computar? Tema 1.3	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
4	- Unidad 1 ¿Qué es computar? Tema 1.4, 1.5	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
5	- Unidad 1 ¿Qué es computar? Tema 1.6	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
6	- Unidad 2 ¿Qué es verdadero? Tema 2.1, 2.2	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
7	- Unidad 2 ¿Qué es verdadero?	Trabajo del estudiante:



Asignatura: Complejidad y Computación
Código: 18775
Centro: Escuela Politécnica Superior
Titulación: Grado en Ingeniería Informática
Nivel: Grado
Tipo: Optativa
Nº de créditos: 6

Semana	Actividad Presencial	Horas no presenciales
	Tema 2.3, 2.4	recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
8	- Unidad 2 ¿Qué es verdadero? Tema 2.5	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
9	- Unidad 2 ¿Qué es verdadero? Tema 2.6	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
10	- Unidad 3 ¿Qué es tratable computacionalmente? Tema 3.1, 3.2, 3.3	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
11	- Unidad 3 ¿Qué es tratable computacionalmente? Tema 3.4, 3.5, 3.6	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
12	- Unidad 4 Más allá de Turing: comput. cuántica Tema 4.1, 4.2	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.



Asignatura: Complejidad y Computación

Código: 18775

Centro: Escuela Politécnica Superior

Titulación: Grado en Ingeniería Informática

Nivel: Grado

Tipo: Optativa

Nº de créditos: 6

Semana	Actividad Presencial	Horas no presenciales
13	- Unidad 4 Más allá de Turing: comput. cuántica Tema 4.3, 4.4	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.
14	- Unidad 4 Más allá de Turing: comput. cuántica Tema 4.5, 4.6	Trabajo del estudiante: recopilación, lectura y análisis crítico de documentación. Presentaciones en clase, Realización de ejercicios, trabajos, y proyectos de programación.