

CENTRO UNIVERSITARIO DE TECNOLOGÍA Y ARTE DIGITAL



PLANIFICACIÓN DE LA DOCENCIA UNIVERSITARIA

GUÍA DOCENTE

LENGUAJES, INTÉRPRETES Y COMPILADORES

1. DATOS DE IDENTIFICACIÓN DE LA ASIGNATURA.

Título:	GRADO DE INGENIERÍA EN DESARROLLO DE CONTENIDOS DIGITALES				
Facultad:	Centro Universitario de Tecnología y Arte Digital (U-TAD)				
Departamento/Instituto:					
Materia:	Fundamentos de Computación				
Denominación de la asignatura:	Lenguajes, Intérpretes y Compiladores				
Código:	0048031				
Curso:	3º				
Semestre:	1º				
Tipo de asignatura (básica, obligatoria u optativa):	Optativa				
Créditos ECTS:	6				
Modalidad/es de enseñanza:	Presencial				
Lengua vehicular:	Español				
Equipo docente:	Pendiente de contratación				
Profesor/a:	Pendiente de contratación				
Grupos:	IDCD3				
Despacho:	Sala de profesores				
Teléfono:	916402811	Ext.	113	E-mail:	--
Página web:	u-tad.blackboard.com				

2. REQUISITOS PREVIOS.

Esenciales:
Conocer conceptos sobre Fundamentos de Programación y Autómatas Formales
Aconsejables:
Haber aprobado las asignaturas de Fundamentos de Programación, Estructuras de los Ordenadores e Introducción a la Teoría de la Computación

3. SENTIDO Y APORTACIONES DE LA ASIGNATURA AL PLAN DE ESTUDIOS.

Campo de conocimiento al pertenece la asignatura.
Esta asignatura pertenece al Módulo de Fundamentos, a la Materia de Fundamentos de Computación.
Relación de interdisciplinariedad con otras asignaturas del curriculum.
Esta asignatura entronca verticalmente con Introducción a la Teoría de la Computación y Estructuras de Ordenadores, y horizontalmente con la asignatura de Programación a bajo Nivel.
Aportaciones al plan de estudios e interés profesional de la asignatura.
La asignatura permite comprender el proceso de compilación que realiza un ordenador sobre un programa informático escrito en cualquier lenguaje, con el objeto de traducirlo al lenguaje máquina ejecutable por el computador.

4. RESULTADOS DE APRENDIZAJE EN RELACIÓN CON LAS COMPETENCIAS QUE DESARROLLA LA ASIGNATURA.

COMPETENCIAS ESPECÍFICAS	RESULTADOS DE APRENDIZAJE RELACIONADOS CON LAS COMPETENCIAS ESPECÍFICAS
<p>CE3 - Demostrar conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería</p> <p>CE9 - Mostrar conocimiento, diseño y aplicación de los procedimientos algorítmicos, tipos y estructuras de datos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.</p>	<p>Conocer los fundamentos de computación y los distintos modelos de computación empleados posteriormente de forma práctica.</p> <p>Emplear la teoría de lenguajes necesaria para construir un lenguaje y la construcción de un sistema de compilación/interpretación.</p>

5. CONTENIDOS / TEMARIO / UNIDADES DIDÁCTICAS

1. Lenguajes regulares y libres de contexto
2. Análisis léxico
3. Análisis sintáctico
4. Análisis semántico
5. Generación de código intermedio
6. Introducción a las optimizaciones
7. Generación de código máquina

6. CRONOGRAMA

UNIDADES DIDÁCTICAS / TEMAS	PERÍODO TEMPORAL
Tema 1	Semana 1
Tema 2	Semanas 2, 3, 4, 5, 6, 7
Tema 3	Semanas 8, 9, 10, 11
Tema 4	Semanas 12, 13, 14

7. MODALIDADES ORGANIZATIVAS Y MÉTODOS DE ENSEÑANZA

MODALIDAD ORGANIZATIVA	MÉTODO DE ENSEÑANZA	COMPETENCIAS RELACIONADAS	HORAS PRESENCIALES	TRABAJO AUTÓNOMO	TOTAL DE HORAS
------------------------	---------------------	---------------------------	--------------------	------------------	----------------

Clases teóricas	Lección magistral	CE3, CE9	29	2	30
Seminarios y talleres	Estudio de casos Resolución de ejercicios y problemas	CE3, CE9	0	0	0
Clases prácticas	Aprendizaje basado en problemas Aprendizaje orientado a proyectos	CE3, CE9	15	0	15
Prácticas externas			0	0	0
Tutorías	Aprendizaje orientado a proyectos Aprendizaje basado en problemas	CE3, CE9	8	0	8
Actividades de evaluación		CE3, CE9	8	0	8
Estudio y trabajo en grupo	Aprendizaje cooperativo	CE3, CE9	1	21	23
Estudio y trabajo autónomo, individual	Estudio de casos Resolución de ejercicios y problemas Aprendizaje basado en problemas Aprendizaje orientado a proyectos	CE3, CE9	0	68	68

8. SISTEMA DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	CRITERIOS DE EVALUACIÓN	VALORACIÓN RESPECTO A LA CALIFICACIÓN FINAL (%)
PR: Prácticas	De 0 a 10 cada uno. Es obligatorio realizar todas las prácticas y obtener una nota media mínima de 5 para hacer media ponderada con examen final	40%
EP: Examen Parcial	De 0 a 10. Es obligatorio obtener al menos un 4 para hacer media ponderada con examen final	20%
EO: Examen final Ordinario	De 0 a 10. Es obligatorio obtener al menos un 4 para aprobar la asignatura.	40%

Consideraciones generales acerca de la evaluación:

Cada semana se dará una hoja de ejercicios para realizarse por el alumno en el tiempo fuera de la clase y entregarse en un plazo máximo de una semana. Los ejercicios serán parecidos a los que se resuelven en la clase como prácticas. La realización de las prácticas es totalmente individual. Las prácticas entregadas fuera del plazo de entrega no se corrigen.

Es obligatorio tener una asistencia superior al 80% de las clases para poder aprobar la asignatura en convocatoria ordinaria.

La nota de los problemas/ejercicios y la del examen parcial sirve tanto para la convocatoria ordinaria como para la extraordinaria, en caso de necesitarse.

Para incluir la nota de evaluación continua en la nota final es obligatorio obtener las siguientes notas:

- al menos un 5 en la nota media de los ejercicios
- al menos un 4 en el examen parcial
- al menos un 4 en la nota del examen final ordinario o un 5 en la nota del examen final extraordinario

En caso de la convocatoria ordinaria, la nota final se calcula como el máximo entre:

- $40\% \times PR + 20\% \times EP + 40\% \times EO$
- $100\% \times EO$.

En caso de la convocatoria extraordinaria, la nota final se calcula como el máximo entre:

- $40\% \times PR + 20\% \times EP + 40\% \times EE$
- $100\% \times EE$

9. BIBLIOGRAFÍA / WEBGRAFÍA

Bibliografía general

1. Compiladores: principios, técnicas y herramientas, 2da Edición, Alfred V. Aho, Monica S. Lam, Ravi Sethi & Jeffrey D. Ullman

Bibliografía recomendada por temas

1. Bennett, J. P.: "Introduction to Compiling Techniques" McGraw-Hill. 1996.
2. Bornat, R.: "Understanding and Writing Compilers: a Do-It-Yourself Guide" MacMillan. 1979.
3. Grune, D.; Bal, H.; Jacobs, C; Langendoen, K.: "Modern Compiler Design" John Wiley & Sons. 2000.

10.- OBSERVACIONES

--