

**CENTRO UNIVERSITARIO DE TECNOLOGÍA Y ARTE
DIGITAL**



PLANIFICACIÓN DE LA DOCENCIA UNIVERSITARIA

GUÍA DOCENTE

INTRODUCCIÓN A LA PROGRAMACIÓN

1. DATOS DE IDENTIFICACIÓN DE LA ASIGNATURA.

Título:	Grado en Diseño de Productos Interactivos				
Facultad:	Centro Universitario de Tecnología y Arte Digital (U-tad)				
Departamento/Instituto:	Área de Arte, Ciencia y Tecnología				
Materia:	Fundamentos de desarrollo				
Denominación de la asignatura:	Introducción a la programación				
Código:	52011				
Curso:	Primero				
Semestre:	Segundo				
Tipo de asignatura (básica, obligatoria u optativa):	Obligatoria				
Créditos ECTS:	3				
Modalidad/es de enseñanza:	Presencial				
Lengua vehicular:	Castellano				
Equipo docente:	Dra. Leila Shafti				
Profesor/a:	Dra. Leila Shafti				
Grupos:	1º DPIN				
Despacho:	Sala de profesores				
Teléfono:	91 640 28 11	Ext.	112	E-mail:	Leila.shafti@live.u-tad.com
Página web: https://www.u-tad.com/					

2. REQUISITOS PREVIOS.

Esenciales:
Haber cursado la asignatura “Álgebra, geometría y física aplicada” y “Tecnología para diseñadores”.
Aconsejables:
No se requieren.

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

Campo de conocimiento al que pertenece la asignatura.
Esta asignatura pertenece al módulo de arte, ciencia y tecnología y dentro de éste a la materia de fundamentos de desarrollo.
Relación de interdisciplinariedad con otras asignaturas del curriculum.
Esta asignatura es continuación de los contenidos teóricos de “Álgebra, geometría y física aplicada” y se basa en el conocimiento teórico de “Tecnología para diseñadores”.
Aportaciones al plan de estudios e interés profesional de la asignatura.
Es una asignatura fundamental para conocer y dominar las bases teórico prácticas que fundan el desarrollo de scripting, prototipado y creación de contenidos digitales. Permitirá mejorar los conocimientos teóricos prácticos de los alumnos en la consecución de sus competencias para la creación de productos de diseño interactivos.

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

COMPETENCIAS GENERALES	RESULTADOS DE APRENDIZAJE RELACIONADOS CON LAS COMPETENCIAS GENERALES
<ul style="list-style-type: none"> • CG1 - Aprender a lo largo de la vida mediante el estudio autónomo y la formación continua. • CG17 - Demostrar habilidad para analizar, sintetizar y recoger información de diversas fuentes. • CG18 - Gestionar adecuadamente la información. 	<ul style="list-style-type: none"> • Desarrollar estrategias de formación continua y autónoma para estar a la vanguardia de las técnicas y procedimientos de la profesión de un diseñador de producto interactivo. • Gestionar correctamente la información analizando, sintetizando y recogiendo información de diversas fuentes. • Gestionar correctamente la información analizando, sintetizando y recogiendo información de diversas fuentes.
COMPETENCIAS ESPECÍFICAS	RESULTADOS DE APRENDIZAJE RELACIONADOS CON LAS COMPETENCIAS ESPECÍFICAS
<ul style="list-style-type: none"> • CE6 Aplicar los fundamentos prácticos de matemáticas y física para la creación de un producto digital interactivo. • CE7 Conocer los fundamentos prácticos del uso y programación de ordenadores y herramientas de desarrollo de productos interactivos. • CE8 Evaluar las implicaciones éticas, técnicas y creativas de la tecnología en el diseño de productos interactivos 	<ul style="list-style-type: none"> • Evaluar las posibilidades y restricciones que impone la tecnología en la construcción del videojuego • Conocer la sintaxis y uso básico de los lenguajes de programación indicados para el diseño de videojuegos • Implementar programas sencillos acompañados de baterías sencillas de pruebas

5. CONTENIDOS

- Fundamentos y metodología de la programación
- Entorno y lenguaje de programación
- Datos y operaciones. Tipos, expresiones y variables.
- Sentencias de control de flujo. Bucles, variables y arrays
- Introducción a la algoritmia

6. CRONOGRAMA

UNIDADES DIDÁCTICAS / TEMAS	PERÍODO TEMPORAL
Fundamentos y metodología de la programación	Semanas 1-2-3
Entorno y lenguaje de programación	Semanas 4-5-6
Datos y operaciones. Tipos, expresiones y variables.	Semanas 7-8-9
Sentencias de control de flujo. Bucles, variables y arrays	Semanas 10-11-12
Introducción a la algoritmia	Semanas 13-14-15

7. MODALIDADES ORGANIZATIVAS Y MÉTODOS DE ENSEÑANZA

MODALIDAD ORGANIZATIVA	MÉTODO DE ENSEÑA	HORAS PRESENCIA	TRABAJO AUTÓNOMO	TOTAL DE HORAS
Clases teóricas	Lección magistral	18	0	18
Seminarios y talleres	Estudio de casos Resolución de ejercicios y problemas	3	0	3
	Aprendizaje basado en problemas Aprendizaje orientado a proyectos	4	0	4
Clases prácticas		0	0	0
Prácticas externas		0	0	0
Tutorías	Aprendizaje orientado a proyectos Aprendizaje basado en problemas	3	0	3
Actividades de evaluación		3	0	3
Estudio y trabajo en grupo	Aprendizaje cooperativo	0	7	7
Estudio y trabajo autónomo, individual	Estudio de casos Resolución de ejercicios y problemas Aprendizaje basado en problemas Aprendizaje orientado a proyectos	0	36	36
Total		32	43	75

8. SISTEMA DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	VALORACIÓN RESPECTO A LA CALIFICACIÓN FINAL (%)
Pruebas de autoevaluación, heteroevaluación o coevaluación	45%
Trabajos, proyectos, informes, memorias, portfolios	50%
Prácticas o pruebas de simulación real o ficticia	5%

9.- BIBLIOGRAFÍA BÁSICA

Bibliografía general

Bibliografía básica:

- *Learning JavaScript, 2nd Edition. Add Sparkle and Life to Your Web Pages.* Shelley Powers. O'Reilly Media. 978-0-596-52187-5, 2008.
- *Getting Started with Processing.* Casey Reas and Ben Fry. Published June 2010, O'Reilly Media.

Bibliografía recomendada por temas

- *The Ruby Programming Language.* David Flanagan, Yukihiro Matsumoto. O'Reilly. 2008. ISBN-10: 0596516177. First Edition.
- *Programming Ruby 1.9 (3rd edition): The Pragmatic Programmers' Guide.* Dave Thomas, with Chad Fowler y Andy Hunt. 2009. ISBN: 978-1-93435-608-1
- *Why's Poignant Guide to Ruby (online).* Jonathan Gillette. <http://mislav.uniqpath.com/poignant-guide/>
- *Thinking in Python.* Bruce Eckel. <http://www.mindview.net/Books/TIPython>
- *Learning Python (Help for Programmers), 4th edition.* Mark Lutz y David Ascher. O'Reilly Media.
- *Starting Out with C++.* Tony Gaddis. Addison-Wesley. ISBN-13: 978-0132576253
- *Jesse Liberty, C++ (Programación).* Anaya Multimedia. ISBN-13: 978-844159793

10.- OBSERVACIONES

Recursos necesarios para la docencia:

- Ordenador del profesor conectado a proyector. Software instalado: Visual Studio.
- Pizarra digital.
- Pizarra blanca.
- Un ordenador para cada alumno en el aula. Software instalado: Visual Studio.

Recursos necesarios para el trabajo personal del alumno, fuera de clase:

- Ordenador de gama doméstica.
- Conexión a internet.
- Software necesario para desarrollar y programar las prácticas que elija el alumno, siempre que vaya en concordancia con los contenidos de la asignatura y la aprobación del profesor.