



DATOS IDENTIFICATIVOS

Informática: Programación I

Asignatura	Informática: Programación I			
Código	O06G151V01103			
Titulación	Grado en Ingeniería Informática			
Descriptor	Creditos ECTS	Selección	Curso	Cuatrimestre
	12	FB	1	1c
Lengua	Castellano			
Impartición	Gallego			
Departamento	Informática			
Coordinador/a	Lado Touriño, María José			
Profesorado	Celard Pérez, Pedro Cuesta Morales, Pedro Lado Touriño, María José Sorribes Fernández, José Manuel Vila Sobrino, Xosé Antón			
Correo-e	mrpepa@uvigo.es			
Web	http://moovi.uvigo.gal			
Descripción general	En esta materia se establecen las bases de la programación estructurada. La base adquirida es imprescindible para poder entender y desarrollar los conocimientos expuestos en numerosas materias a lo largo de los estudios y en la vida profesional. Cualquiera de los tres perfiles profesionales que recogen los ámbitos de actuación más comunes de las/de los ingenieras/os en Informática de hoy en día contempla la necesidad de poseer competencias relativas al desarrollo e implementación del software. Parte del material didáctico puede estar en inglés.			

Competencias

Código	
A2	Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.
B8	Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.
B9	Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.
C3	Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería
C4	Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
C5	Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería
C12	Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos
C13	Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema
D5	Capacidad de organización y planificación
D7	Capacidad de buscar, relacionar y estructurar información proveniente de diversas fuentes y de integrar ideas y conocimientos.
D8	Capacidad de trabajar en situaciones de falta de información y/o bajo presión

Resultados de aprendizaje

Resultados previstos en la materia	Resultados de Formación y Aprendizaje
------------------------------------	---------------------------------------

RA1: Adquirir las habilidades básicas para analizar un problema y conseguir desarrollar un programa en un lenguaje de alto nivel que permita solucionarlo.	A2	B8	C3 C5 C12 C13	D5 D7 D8
RA2: Adquirir los conocimientos básicos de programación, independientes del lenguaje de programación utilizado.		B8	C3 C4 C5 C12 C13	D5 D7 D8
RA3: Adquirir buenos hábitos de programación, primando la sencillez y legibilidad de los programas, así como realizando, como paso previo a la programación, un análisis de la solución.		B8	C3 C5 C12 C13	D5 D7
RA4: Adquirir un conocimiento detallado y práctico de las características y recursos del lenguaje de programación utilizado en el módulo.	A2	B8	C4 C5	D5 D7 D8
RA5: Usar las herramientas de un entorno de desarrollo de programación para crear y desarrollar aplicaciones.	A2	B8 B9	C4	D8

Contenidos

Tema

1. Algoritmos y programas	I. Elementos de un programa: datos y algoritmos II. Codificación de la información en memoria III. Lenguajes de programación IV. Lenguaje máquina y ensamblador V. Lenguajes de alto nivel VI. Compilación vs. interpretación de programas VII. Paradigmas de programación: imperativa, lógica y funcional
2. Metodología de la programación	I. Especificación de algoritmos II. Diseño de algoritmos II.1. Diagramas de flujo II.2. Pseudocódigo III. Codificación y prueba IV. Compilación y ejecución V. Documentación y mantenimiento
3. Variables e instrucciones	I. Estructura de un programa II. Palabras reservadas e identificadores III. Variables, constantes y tipos de datos simples IV. Instrucciones de asignación V. Expresiones aritméticas y lógicas VI. Instrucciones de Entrada/Salida VII. Estructuras de control
4. Programación estructurada	I. Teorema de la programación estructurada II. Diseño descendente
5. Programación modular	I. Funciones y procedimientos II. Declaración y llamada de funciones III. Paso de parámetros IV. Variables locales y globales V. Diseño modular V.1. Divide y vencerás V.2. Backtracking VI. Recursividad VII. Bibliotecas
6. Depuración y Pruebas	I. Errores II. Pruebas
7. Estructuras y uniones	I. Estructuras II. Uniones III. Operaciones IV. Estructuras como parámetros
8. Arrays	I. Definición II. Vectores III. Matrices IV. Arrays multidimensionales V. Arrays como parámetros
9. Ficheros	I. Tipos de acceso: secuencial y directo II. Operaciones con ficheros III. Funciones de tratamiento de ficheros

10. Gestión dinámica de memoria	I. Concepto de puntero II. Asignación y liberación de memoria III. Operaciones con punteros IV. Punteros y funciones V. Punteros y estructuras VI. Punteros y arrays VII. Arrays dinámicos
11. Cadenas	I. Lectura y escritura II. Asignación III. Operaciones

Planificación

	Horas en clase	Horas fuera de clase	Horas totales
Resolución de problemas	64.5	108.5	173
Estudio previo	0	75.5	75.5
Lección magistral	19.5	19.5	39
Examen de preguntas objetivas	0	5.5	5.5
Resolución de problemas y/o ejercicios	7	0	7

*Los datos que aparecen en la tabla de planificación son de carácter orientativo, considerando la heterogeneidad de alumnado

Metodologías

	Descripción
Resolución de problemas	Actividad en la que se formulan problemas y/o ejercicios relacionados con la materia. El alumnado debe desarrollar las soluciones idóneas o correctas mediante la aplicación de algoritmos. El objetivo es que el alumnado aplique los contenidos teóricos en la resolución de pequeños problemas de programación.
Estudio previo	Búsqueda, lectura y trabajo de documentación, previo a las clases de aula, que realiza el alumnado de forma autónoma.
Lección magistral	Exposición por parte del profesorado de los contenidos sobre la materia objeto de estudio, bases teóricas y directrices. El profesorado podrá solicitar la participación activa del alumnado.

Atención personalizada

Metodologías	Descripción
Resolución de problemas	Actividad académica desarrollada por el profesorado, que tiene como finalidad atender las necesidades y consultas del alumnado relacionadas con el estudio y/o temas vinculados con la materia, proporcionándole orientación, apoyo y motivación en el proceso de aprendizaje. Las sesiones de tutorización podrán realizarse por medios telemáticos (correo electrónico, videoconferencia, foros de FAITIC, ...) bajo la modalidad de concertación previa.
Estudio previo	Actividad académica desarrollada por el profesorado, que tiene como finalidad atender las necesidades y consultas del alumnado relacionadas con el estudio y/o temas vinculados con la materia, proporcionándole orientación, apoyo y motivación en el proceso de aprendizaje. Las sesiones de tutorización podrán realizarse por medios telemáticos (correo electrónico, videoconferencia, foros de FAITIC, ...) bajo la modalidad de concertación previa.

Evaluación

	Descripción	Calificación	Resultados de Formación y Aprendizaje
Examen de preguntas objetivas	Pruebas que incluyen preguntas con diferentes alternativas de respuesta, con las que se pretende comprobar se se alcanzaron las competencias de la materia. Se hará una prueba por cada tema. Resultados de aprendizaje evaluados: RA1, RA2, RA4, RA5.	20 A2	B8 C3 D5 B9 C4 D7 C5 D8 C12 C13
Resolución de problemas y/o ejercicios	Pruebas de programación con las que se pretende comprobar si el alumnado alcanzó las competencias de la materia. - Contenidos teóricos (2 pruebas, 15% cada una). - Contenidos prácticos (2 pruebas, 20% cada una). - Proyecto personal (10%) Resultados de aprendizaje evaluados: RA1, RA2, RA3, RA4, RA5.	80 A2	B8 C3 D5 B9 C4 D7 C5 D8 C12 C13

Otros comentarios sobre la Evaluación

EMPLEO DE DISPOSITIVOS MÓVILES

Se recuerda a todo el alumnado la prohibición del uso de dispositivos móviles o electrónicos y ordenadores portátiles en ejercicios y prácticas, en cumplimiento del artículo 13.2.d) del Estatuto del Estudiante Universitario, relativo a los deberes del estudiantado universitario, que establece el deber de "Abstenerse de la utilización o la cooperación en procedimientos fraudulentos en las pruebas de evaluación, en los trabajos que se realicen o en documentos oficiales de la universidad".

CRITERIOS DE EVALUACIÓN - PRIMERA EDICIÓN DE ACTAS

• **ALUMNADO ASISTENTE**

El alumnado que realice alguna actividad evaluable, cualquiera que sea el tipo, seguirá el procedimiento de evaluación con las metodologías reflejadas en la tabla indicada en el apartado de evaluación. Además, deberá subir obligatoriamente una foto tipo carné al perfil de la plataforma Faitic al principio del curso.

Si un/a estudiante abandona la evaluación continua para asistentes habiendo sido ya evaluado/a de algún contenido de la materia, se considerará que tiene suspensa la convocatoria, y no podrá optar en la misma por la modalidad de no asistente.

• **ALUMNADO NO ASISTENTE**

- **Examen de preguntas objetivas:** prueba que incluye preguntas con diferentes alternativas de respuesta, con la que se pretende comprobar si alcanzaron las competencias de la materia.

- *Porcentaje en la calificación:* 20%.

- *Competencias evaluadas:* CB2, CG8, CG9, CE3, CE4, CE5, CE7, CE12, CT5, CT7, CT8.

- *Resultados de aprendizaje evaluados:* RA1, RA2, RA4, RA5.

- **Resolución de problemas y/o ejercicios:** prueba en la que el alumnado debe solucionar una serie de problemas y/o ejercicios en un tiempo/condiciones establecido/as, con la que se pretende comprobar si se alcanzaron las competencias de la materia. Dos pruebas:

1. Contenidos teóricos (30%).
2. Contenidos prácticos (50%).

- *Porcentaje en la calificación:* 80%.

- *Competencias evaluadas:* CB2, CG8, CG9, CE3, CE4, CE5, CE7, CE12, CT5, CT7, CT8.

- *Resultados de aprendizaje evaluados:* RA1, RA2, RA3, RA4, RA5.

CRITERIOS DE EVALUACIÓN - SEGUNDA EDICIÓN DE ACTAS Y FIN DE CARRERA

Se empleará el mismo sistema de evaluación aplicado para alumnado no asistente.

PROCESO DE CALIFICACIÓN DE ACTAS

Independientemente de la convocatoria, para superar la materia es **IMPRESINDIBLE** sacar una puntuación igual o superior a 5 sobre 10 en todas y cada una de las partes que intervienen en la evaluación. En caso de que la calificación resultante sea igual o superior a 5, pero alguna de las partes esté suspensa, la calificación final máxima podrá ser de hasta 4 (SUSPENSO).

FECHAS OFICIALES DE PRUEBAS DE EVALUACIÓN

El calendario de pruebas de evaluación aprobado oficialmente por la Xunta de Centro de la ESEI se encuentra publicado en

Fuentes de información

Bibliografía Básica

JOYANES AGUILAR, Luis, **Fundamentos de Programación**, 84-481-6111-4, 4ª, McGraw Hill, 2008

JOYANES AGUILAR, Luis, RODRÍGUEZ BAENA, Luis, FERNÁNDEZ AZUELA, Matilde, **Fundamentos de Programación. Libro de Problemas**, 84-481-3986-0, 2ª, McGraw Hill, 200

JOYANES AGUILAR, Luis, ZAHONERO MARTÍNEZ, Ignacio, **Programación en C: Metodología, algoritmos y estructuras de datos**, 84-481-9844-1, 2ª, McGraw Hill, 84-481-9844-1

RODRÍGUEZ RANCEL, Mario, **Aprende a programar con pseudocódigo, diagramas de flujo y ejercicios ejemplo resueltos en C**, 978-84-939427-7-9, 1ª, aprenderaprogramar.com, 2015

Bibliografía Complementaria

BROOKSHEAR, J. Glenn, **Introducción a la Computación**, 978-84-7829-139-7, 12ª, Pearson Educación, 2013

CEBALLOS SIERRA, Francisco Javier, **C/C++ Curso de Programación**, 978-84-9964-812-5, 5ª, Ra-Ma, 2019

KERNIGHAN, Brian W., RITCHIE, Dennis M, **El lenguaje de programación C**, 968-880-205-0, 2ª, Pearson Educación, 1991

PRIETO ESPINOSA, Alberto, LLORIS RUIZ Antonio, TORRES CANTERO Juan Carlos, **Introducción a la Informática**, 84-481-4624-7, 4ª, McGraw Hill, 2006

VIRGÓS BEL, Ferrán; SEGURA CASANOVA, Joan, **Fundamentos de informática: En el marco del espacio europeo de enseñanza superior**, 84-481-6747-3, 1ª, McGraw Hill, 2008

Recomendaciones

Asignaturas que continúan el temario

Informática: Algoritmos y estructuras de datos I/O06G151V01107

Programación II/O06G151V01109

Otros comentarios

El/la estudiante debe preparar la materia, consultando la bibliografía y asistiendo con regularidad a las sesiones prácticas. Debido al carácter práctico de la materia, se recomienda que se realicen todas las actividades propuestas.

Plan de Contingencias

Descripción

=== MEDIDAS EXCEPCIONALES PLANIFICADAS ===

ESCENARIO 1: DOCENCIA MIXTA

Debido a la situación excepcional, ante la imposibilidad de poder impartir la docencia de un modo presencial, se utilizarán medios virtuales para la impartición de las clases.

ESCENARIO 2: DOCENCIA NO PRESENCIAL

Debido a la situación excepcional, ante la imposibilidad de poder impartir la docencia de un modo presencial, se utilizarán medios virtuales para la impartición de las clases.

En ambos casos, la actividad docente se impartirá mediante Campus Remoto y se preverá también el uso de la plataforma de teledocencia Faitic como refuerzo y sin perjuicio de otras medidas que se puedan adoptar para garantizar la accesibilidad del alumnado a los contenidos docentes.

Además, todo el material estará disponible en la plataforma de teledocencia Faitic para facilitar al máximo la organización del trabajo de forma autónoma por parte del alumnado, en previsión de problemas de conciliación y/o conectividad.