

GUIA DOCENTE DE LA ASIGNATURA

Fundamentos de Programación

MÓDULO	MATERIA	CURSO	SEMESTRE	CRÉDITOS	TIPO
Formación Básica	Informática	1	1	6	Obligatoria
PROFESOR(ES)		DIRECCIÓN COMPLETA DE CONTACTO PARA TUTORÍAS (Dirección postal, teléfono, correo electrónico, etc.)			
Teoría: Juan Carlos Cubero Prácticas: Juan Carlos Cubero		Departamento de Ciencias de la Computación e I.A. E.T.S.I.I.T. - Universidad de Granada C/Daniel Saucedo Aranda s/n 18071-GRANADA Teléfono: 958244019; Fax: 948243317 http://decsai.ugr.es			
		HORARIO DE TUTORÍAS			
		Lunes: 10-13h, Martes: 10-13h			
GRADO EN EL QUE SE IMPARTE		OTROS GRADOS A LOS QUE SE PODRÍA OFERTAR			
Doble grado en Informática y Matemáticas					
PRERREQUISITOS Y/O RECOMENDACIONES (Si ha lugar)					



BREVE DESCRIPCIÓN DE CONTENIDOS (SEGÚN MEMORIA DE VERIFICACIÓN DEL GRADO)

Tipos de datos básicos. Objetos, operadores y expresiones. Estructuras de control. Funciones y procedimientos. Tipos de datos compuestos: homogéneos (arrays) y heterogéneos (registros). Algoritmos básicos de ordenación y búsqueda. Recursividad.

COMPETENCIAS GENERALES Y ESPECÍFICAS**Competencias Específicas de la Asignatura**

B4. Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

B5. 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.

Competencias Específicas del Título

E8. 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.

E9. 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.

Competencias Transversales o Generales

T1. Capacidad de análisis y síntesis: Encontrar, analizar, criticar (razonamiento crítico), relacionar, estructurar y sintetizar información proveniente de diversas fuentes, así como integrar ideas y conocimientos.

T2. Capacidad de organización y planificación así como capacidad de gestión de la Información

T3. Capacidad de comunicación oral y escrita en el ámbito académico y profesional con especial énfasis, en la redacción de documentación técnica

T4. Capacidad para la resolución de problemas

T5. Capacidad para tomar decisiones basadas en criterios objetivos (datos experimentales, científicos o de simulación disponibles) así como capacidad de argumentar y justificar lógicamente dichas decisiones, sabiendo aceptar otros puntos de vista

T6. Capacidad para el uso y aplicación de las TIC en el ámbito académico y profesional.

T7. Capacidad de comunicación en lengua extranjera, particularmente en inglés.

T8. Capacidad de trabajo en equipo.

T9. Capacidad para el aprendizaje autónomo así como iniciativa y espíritu emprendedor

T10. Motivación por la calidad y la mejora continua, actuando con rigor, responsabilidad y ética profesional.



T11. Capacidad para adaptarse a las tecnologías y a los futuros entornos actualizando las competencias profesionales.

T12. Capacidad para innovar y generar nuevas ideas.

T13. Sensibilidad hacia temas medioambientales

T14. Respeto a los derechos fundamentales y de igualdad entre hombres y mujeres

T15. Capacidad para proyectar los conocimientos, habilidades y destrezas adquiridos para promover una sociedad basada en los valores de la libertad, la justicia, la igualdad y el pluralismo



ugr

Universidad
de Granada

INFORMACIÓN SOBRE TITULACIONES DE LA UGR
<http://grados.ugr.es>

OBJETIVOS (EXPRESADOS COMO RESULTADOS DE APRENDIZAJE)

Comprender el funcionamiento de un computador, haciendo especial énfasis en la necesidad de desarrollo de software por parte del programador.

Presentar la historia de la programación y de los distintos paradigmas de programación, situando en ese contexto el lenguaje de programación que se va a utilizar.

Comprender la necesidad de un proceso de traducción de un lenguaje de alto nivel.

Conocer y distinguir los conceptos de algoritmo y programa

Mostrar la necesidad de codificar la información que maneja internamente un computador, enfatizando posibilidades y limitaciones cuando se resuelve un problema.

Conocer los tipos de datos primitivos y sus operaciones.

Distinguir entre tipo de dato y objeto.

Conocer las acciones básicas de E/S de datos.

Aprender a usar las estructuras de control básicas: secuencial, condicional e iterativa.

Comprender la necesidad de dividir la solución creando módulos (funciones o procedimientos) que implementen operaciones no primitivas.

Comprender la necesidad de la especificación de una función o procedimiento, como método de abstracción, introduciendo los conceptos de precondition y postcondition.

Aprender a resolver problemas aplicando una metodología de diseño modular (top-down/bottom-up).

Manejar correctamente los mecanismos de comunicación entre módulos (interfaces), así como los distintas formas de paso de parámetros y devolución de resultados.

Entender la gestión de llamadas a funciones mediante la pila.

Motivar e introducir los tipos de datos compuestos, arrays y registros, así como sus operaciones.

Conocer los algoritmos de ordenación básicos (selección, inserción, burbuja).

Conocer los algoritmos de búsqueda básicos (lineal, dicotómica).

Motivar y aprender a resolver problemas mediante algoritmos recursivos.



TEMARIO DETALLADO DE LA ASIGNATURA**TEMARIO TEÓRICO****Tema 1. Programación en C++: Fundamentos**

- 1.1 El ordenador, algoritmos y programas
 - 1.1.1 El Ordenador: Conceptos Básicos
 - 1.1.2 Lenguajes de programación
 - 1.1.3 Algoritmos
 - 1.1.4 Compilación
- 1.2 Especificación de programas
 - 1.2.1 Escritura y organización de un programa
 - 1.2.2 Elementos básicos de un lenguaje de programación
 - 1.2.3 Tipos de errores en la programación
- 1.3 Datos y tipos de datos
 - 1.3.1 Representación en memoria de datos e instrucciones
 - 1.3.2 Datos y tipos de datos
- 1.4 Operadores y Expresiones
 - 1.4.1 Operadores en un lenguaje de programación
 - 1.4.2 Expresiones
- 1.5 Tipos de datos comunes en C++
 - 1.5.1 Rango y operadores aplicables a un tipo de dato
 - 1.5.2 Los tipos de datos básicos
 - 1.5.3 El tipo enumerado y el tipo registro -struct-

Tema 2. Estructuras de Control

- 2.1 Estructura Condicional
 - 2.1.1 Introducción
 - 2.1.2 Estructura Condicional Simple
 - 2.1.3 Estructura condicional Doble
 - 2.1.4 Anidamiento de Estructuras Condicionales
 - 2.1.5 Estructura Condicional Múltiple
- 2.2 Estructuras Repetitivas
 - 2.2.1 Bucles controlados por condición: pre-test y post-test
 - 2.2.2 Bucles controlados por contador
 - 2.2.3 Anidamiento de bucles

Tema 3. Funciones y Clases

- 3.1 Funciones
 - 3.1.1 Introducción a las funciones
 - 3.1.2 Parámetros y datos locales
 - 3.1.3 Funciones void
 - 3.1.4 Sobrecarga de funciones
- 3.2 Clases
 - 3.2.1 Definición de una clase y creación de objetos
 - 3.2.2 Encapsulamiento: variables de instancia y métodos
 - 3.2.3 Comunicación entre objetos

Tema 4. Vectores y Matrices

- 4.1 Vectores
 - 4.1.1 Declaración y Representación en Memoria



ugr

Universidad
de Granada

INFORMACIÓN SOBRE TITULACIONES DE LA UGR
<http://grados.ugr.es>

- 4.1.2 Operaciones con Vectores
- 4.1.3 Clases y Vectores
- 4.1.4 Algoritmos de Búsqueda
- 4.1.5 Algoritmos de Ordenación
- 4.2 Matrices
 - 4.2.1 Declaración y Operaciones con matrices
 - 4.2.2 Matrices de varias dimensiones
- Tema 5. Recursividad
 - 5.1 Diseño de algoritmos recursivos
 - 5.2 Funciones recursivas

TEMARIO PRÁCTICO

Práctica 1.

Realización de Problemas sobre:

- Construcción de expresiones, sentencias y programas
- Resolución de problemas con algoritmos
- Uso de tipos de datos para representar información

Presentación y uso de un entorno de desarrollo para programar en C++.

Práctica 2. Realización de Problemas sobre:

- Estructuras condicionales
- Estructuras repetitivas

Práctica 3.

Realización de Problemas sobre:

- Definición y uso de funciones que trabajan sobre tipos simples
- Definición de clases. Ámbitos privado y público
- Acceso de los métodos de una clase a sus variables de instancia

Proyecto 1. Planteamiento de un conjunto de pequeños proyectos de programación.

Práctica 4. Realización de Problemas sobre:

- Construcción de clases que proporcionen acceso y operaciones sobre conjunto de datos
- Otros métodos de búsqueda y ordenación de vectores



Práctica 5.

Realización de Problemas sobre Recursividad.

Proyecto 2. Planteamiento de un conjunto de pequeños proyectos de programación.

SEMINARIOS

Seminario 1. La riqueza de los tipos de datos en C++.

- Operaciones con tipos numéricos distintos
- Calificadores de signo y de tamaño

Seminario 2. Particularidades de las estructuras de control en C++

- Las variantes del bucle for en C++
- Cuestiones de estilo
- Otras (perniciosas) estructuras de control

Seminario 3. Documentación de funciones.

- Documentación de cabeceras
- Descripción de algoritmos

Seminario 4. La necesidad de la gestión de errores.

Seminario 5. Las clases estándar Vector y String

- La clase Vector
- La clase String



BIBLIOGRAFÍA

BIBLIOGRAFÍA FUNDAMENTAL:

Deitel & Deitel. C++ Cómo Programar. Sexta edición, Prentice Hall-Pearson, 2008
 Garrido, A. Fundamentos de Programación en C++. Delta Publicaciones, 2005.
 Mercer, R., Computing Fundamentals with C++. Object-oriented Programming and Design. 2ª Edición. Palgrave Macmillan. 2000.
 Walter Savitch. Resolución de problemas con C++, Pearson, 2006.

BIBLIOGRAFÍA COMPLEMENTARIA:

Robert Martin. Clean Code: A Handbook of Agile Software Craftsmanship, Prentice Hall, 2008
 Steve McConnell. Code Complete: A Practical Handbook of Software Construction. Microsoft Press; 2nd edition, 2004
 Stephen Prata. C++ Primer Plus. Fifth edition. SAMS, 2004
 Bjarne Stroustrup. El Lenguaje de Programación C++. Addison Wesley, 2001.

ENLACES RECOMENDADOS

C++ con clase. <http://c.conclase.net>
 C Plus Plus (en inglés) <http://www.cplusplus.com>
 C++ Reference (en inglés) <http://www.cppreference.com>
 Zator (libro programación) <http://www.zator.com/Cpp/>
 Plataforma enseñanza C++. <http://decsai.ugr.es/~jfv/ed1/c%2B%2B/cdrom3/TIC-CD/web/index.htm>



PROGRAMA DE ACTIVIDADES								
Primer cuatrimestre	Actividades presenciales							Actividades no presenciales
	Temas	Sesiones teóricas (horas)	Sesiones prácticas (horas)	Exposiciones y seminarios (horas)	Visitas y excursiones (horas)	Exámenes	Tutorías grupales (horas)	
Semana 1	Tema 1 Práctica 1	2	2					4
Semana 2	Tema 1 Práctica 1	2	1				1	4
Semana 3	Temas 1 y 2 Seminario 1	2		2				4
Semana 4	Tema 2 Práctica 2	2	2					4
Semana 5	Tema 2 Práctica 2	2	1				1	4
Semana 6	Temas 2 y 3 Seminario 2	2		2				4
Semana 7	Tema 3 Práctica 3	1	2			1		4
Semana 8	Tema 3 Práctica 3	2	1				1	4
Semana 9	Tema 3 Seminario 3	2		2				4
Semana 10	Tema 4 Práctica 4	2	2					4
Semana 11	Tema 4 Práctica 4	2	1				1	4
Semana 12	Tema 4 Seminario 4	1		2				4
Semana 13	Tema 4 Seminario 5	1		2				4
Semana 14	Tema 5 Práctica 5	1	1			1	1	4
Semana 15	Tema 5 Práctica 5	2	2					4
Resto						2		30
Total horas								90

METODOLOGÍA DOCENTE
1. Lección magistral (Clases teóricas-expositivas) (grupo grande) Descripción: Presentación en el aula de los conceptos propios de la materia haciendo uso de metodología expositiva con lecciones magistrales participativas y medios audiovisuales. Evaluación y examen de las capacidades adquiridas. Propósito: Transmitir los contenidos de la materia motivando al alumnado a la reflexión, facilitándole el descubrimiento de las relaciones entre diversos conceptos y formarle una mentalidad crítica Contenido en ECTS: 30 horas presenciales (1.2 ECTS) Competencias: E8, E9, B4, B5, T1, T2, T3, T4, T5, T6, T7, T10, T11, T12, T13, T14, T15



2. Actividades prácticas (Clases prácticas de laboratorio) (grupo pequeño)

Descripción: Actividades a través de las cuales se pretende mostrar al alumnado cómo debe actuar a partir de la aplicación de los conocimientos adquiridos

Propósito: Desarrollo en el alumnado de las habilidades instrumentales de la materia.

Contenido en ECTS: 15 horas presenciales (0.6 ECTS)

Competencias: E8, E9, B4, B5, T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11, T12, T13, T14, T15

3. Seminarios (grupo pequeño)

Descripción: Modalidad organizativa de los procesos de enseñanza y aprendizaje donde tratar en profundidad una temática relacionada con la materia. Incorpora actividades basadas en la indagación, el debate, la reflexión y el intercambio.

Propósito: Desarrollo en el alumnado de las competencias cognitivas y procedimentales de la materia.

Contenido en ECTS: 10 horas presenciales (0.4 ECTS)

Competencias: B4, B5, E8, T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11, T12, T13, T14, T15

4. Actividades no presenciales individuales (Estudio y trabajo autónomo)

Descripción: 1) Actividades (guiadas y no guiadas) propuestas por el profesor a través de las cuales y de forma individual se profundiza en aspectos concretos de la materia posibilitando al estudiante avanzar en la adquisición de determinados conocimientos y procedimientos de la materia, 2) Estudio individualizado de los contenidos de la materia 3) Actividades evaluativas (informes, exámenes, ...)

Propósito: Favorecer en el estudiante la capacidad para autorregular su aprendizaje, planificándolo, diseñándolo, evaluándolo y adecuándolo a sus especiales condiciones e intereses.

Contenido en ECTS: 45 horas no presenciales (1.8 ECTS)

Competencias: B4, B5, E8, E9, T1, T2, T3, T4, T5, T6, T7, T9, T10, T11, T12, T13, T14, T15

5. Actividades no presenciales grupales (Estudio y trabajo en grupo)

Descripción: Actividades (guiadas y no guiadas) propuestas por el profesor a través de las cuales y de forma grupal se profundiza en aspectos concretos de la materia posibilitando a los estudiantes avanzar en la adquisición de determinados conocimientos y procedimientos de la materia.

Propósito: Favorecer en los estudiantes la generación e intercambio de ideas, la identificación y análisis de diferentes puntos de vista sobre una temática, la generalización o transferencia de conocimiento y la valoración crítica del mismo.

Contenido en ECTS: 45 horas no presenciales (1.8 ECTS)

Competencias: B4, B5, T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11, T12, T13, T14, T15

6. Tutorías académicas (grupo pequeño)

Descripción: manera de organizar los procesos de enseñanza y aprendizaje que se basa en la interacción directa entre el estudiante y el profesor

Propósito: 1) Orientan el trabajo autónomo y grupal del alumnado, 2) profundizar en distintos aspectos de la materia y 3) orientar la formación académica-integral del estudiante

Contenido en ECTS: 5 horas presenciales, grupales e individuales (0.2 ECTS)

Competencias: E8, E9, T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11, T12, T13, T14, T15

EVALUACIÓN (INSTRUMENTOS DE EVALUACIÓN, CRITERIOS DE EVALUACIÓN Y PORCENTAJE SOBRE LA CALIFICACIÓN FINAL, ETC.)



ugr

Universidad
de Granada

INFORMACIÓN SOBRE TITULACIONES DE LA UGR

<http://grados.ugr.es>

- Para la parte teórica se realizará un examen escrito multipregunta sobre los contenidos de la materia impartida. La ponderación de este bloque es el 60%.
- Para la parte práctica se propondrán la realización de diversos proyectos individuales que se evaluarán en dos sesiones a lo largo de la asignatura. La ponderación del primer proyecto será del 10% y la del segundo un 20%, llegando por tanto la parte práctica a un total del 30%.
- La evaluación del trabajo autónomo se realizará de forma continua. Los alumnos deben formar grupos de trabajo que pueden variar, si así lo desean, a lo largo del curso. Los grupos desarrollarán un guión de trabajo entregado con anterioridad y lo defenderán durante las sesiones de tutorías académicas. La ponderación de esta parte es del 10%.

La calificación global corresponderá por tanto a la puntuación ponderada de los diferentes aspectos y actividades que integran el sistema de evaluación. Por tanto, el resultado de la evaluación será una calificación numérica obtenida mediante la suma ponderada de las calificaciones correspondientes a una parte teórica, una parte práctica y, en su caso, una parte relacionada con el trabajo autónomo de los alumnos, los seminarios impartidos y el aprendizaje basado en proyectos.

Todo lo relativo a la evaluación se regirá por la normativa sobre planificación docente y organización de exámenes vigente en la Universidad de Granada.

El sistema de calificaciones se expresará mediante calificación numérica de acuerdo con lo establecido en el art. 5 del R. D 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de calificaciones en las titulaciones universitarias de carácter oficial y validez en el territorio nacional.

INFORMACIÓN ADICIONAL

Plataforma docente usada:
<http://decsai.ugr.es/fp>

Definición de grupo grande y grupo pequeño:
 Los grupos grandes son grupos de 45 a 60 estudiantes.
 Los grupos pequeños son grupos de 15 a 20 estudiantes.



ugr | Universidad
de Granada

INFORMACIÓN SOBRE TITULACIONES DE LA UGR
<http://grados.ugr.es>