GUÍA DOCENTE

DENOMINACIÓN DE LA ASIGNATURA

Denominación: **METODOLOGÍA DE LA PROGRAMACIÓN**
Código: 101381
Plan de estudios: **GRADO DE INGENIERÍA INFORMÁTICA** Curso: 1
Denominación del módulo al que pertenece: FORMACIÓN BÁSICA
Materia: INFORMÁTICA
Carácter: BASICA
Duración: SEGUNDO CUATRIMESTRE
Créditos ECTS: 6.0
Horas de trabajo presencial: 60
Porcentaje de presencialidad: 40.0%
Horas de trabajo no presencial: 90
Plataforma virtual: https://moodle.uco.es/

DATOS DEL PROFESORADO

Nombre: GIBAJA GALINDO, EVA LUCRECIA (Coordinador)
Departamento: INFORMÁTICA Y ANÁLISIS NUMÉRICO
Área: CIENCIA DE LA COMPUTACIÓN E INTELIGENCIA ARTIFICIAL
Ubicación del despacho: Edificio Albert Einstein (C2), tercera planta
E-Mail: in1gigae@uco.es Teléfono: 957212031

Nombre: FERNÁNDEZ GARCÍA, NICOLÁS LUIS
Departamento: INFORMÁTICA Y ANÁLISIS NUMÉRICO
Área: CIENCIA DE LA COMPUTACIÓN E INTELIGENCIA ARTIFICIAL
Ubicación del despacho: Edificio Albert Einstein (C2), tercera planta
E-Mail: ma1fegan@uco.es Teléfono: 957218347

Nombre: MENDOZA HURTADO, MANUEL
Departamento: INFORMÁTICA Y ANÁLISIS NUMÉRICO
Área: CIENCIA DE LA COMPUTACIÓN E INTELIGENCIA ARTIFICIAL
Ubicación del despacho: Edificio Albert Einstein (C2), tercera planta
E-Mail: i52mehum@uco.es Teléfono: 957212031

Nombre: VARGAS YUN, VÍCTOR MANUEL
Departamento: INFORMÁTICA Y ANÁLISIS NUMÉRICO
Área: CIENCIA DE LA COMPUTACIÓN E INTELIGENCIA ARTIFICIAL
Ubicación del despacho: Edificio Albert Einstein (C2), tercera planta
E-Mail: i42vayuv@uco.es Teléfono: 957212031

GUÍA DOCENTE

REQUISITOS Y RECOMENDACIONES

Requisitos previos establecidos en el plan de estudios

Se recomienda al alumno un buen dominio de los conceptos vistos en la asignatura Introducción a la Programación:
+Tipos de datos simples: enteros, float, char
+Entrada y salida de datos (printf y scanf)
+Esquemas condicional e iterativo
+Funciones y paso de parámetros
+Tipos de datos compuestos: vectores, matrices, cadenas y estructuras

También es recomendable manejar con soltura la terminal del sistema.

Recomendaciones

*Sería interesante que el alumno dispusiera de un ordenador y un entorno de programación completo y actualizado en su casa para ampliar y poner en práctica los conocimientos recibidos en asignatura.
*Se recomienda al alumno seguir la asignatura desde el primer día, repasando en casa los conceptos vistos en clase y programando los ejemplos y ejercicios propuestos.
*Cualquier ambigüedad en la guía será resuelta por la profesora coordinadora de la asignatura.
*La metodología docente propuesta considera el porcentaje de presencialidad de la asignatura indicado en el plan de estudios, según el cual la asignatura contempla 60 horas de trabajo presencial y 90 horas de trabajo no presencial. Por tanto, para conseguir un adecuado aprovechamiento del curso, adquirir las competencias y superar la asignatura, se debe asistir a clase y realizar regularmente trabajo personal en casa.

COMPETENCIAS

CB4	Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.
CU2	Conocer y perfeccionar el nivel de usuario en el ámbito de las TIC.
CEB4	Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
CEB5	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.

OBJETIVOS

El objetivo general es proporcionar una importante base teórica y práctica en el aprendizaje de la programación de ordenadores. Esta base permitirá cursar posteriores. En términos de recursos de aprendizaje, se espera que, tras superar la asignatura el alumno:
*Maneja correctamente tipos de datos compuestos: estructuras, arrays y cadenas.
*Utiliza adecuadamente el tipo de dato puntero: declaración, operadores, paso de parámetros, *arrays*, punteros genéricos (*void**) y punteros a funciones.
*Conoce la estructura de un programa en tiempo de ejecución: código, datos, *stack* y *heap*, relacionándola con su aplicación en entornos de programación.
*Gestiona adecuadamente la memoria dinámica de un programa haciendo un uso adecuado de las funciones de reserva y liberación proporcionadas por el lenguaje.
*Entiende el concepto de fichero como método de almacenamiento persistente, diferenciando entre ficheros de texto y binarios y acceso secuencial y aleatorio.

GUÍA DOCENTE

*Utiliza adecuadamente las funciones proporcionadas por el lenguaje para lectura/escritura de ficheros binarios y de texto construyendo programas capaces de procesar ficheros (recorrido, actualización y borrado) de forma correcta.
*Comprende y maneja el concepto de recursividad construyendo algoritmos y programas recursivos.
*Maneja adecuadamente los conceptos de lista, pila y colas siendo capaz de implementar funciones para su procesamiento.
*Conoce el funcionamiento de algoritmos básicos de búsqueda y ordenación y su complejidad algorítmica, siendo capaz de implementarlos en un programa.
*Aplica pautas metodológicas al desarrollo de programas y utiliza herramientas tales como el depurador, el preprocesador (inclusión condicional de código), bibliotecas, generación automática de proyectos (ej. *makefiles*), generadores de documentación, o el mecanismo de paso de parámetros al programa principal.
*Conoce el objetivo de las pruebas y realiza programas correctos, claros, y eficientes.

CONTENIDOS

1. Contenidos teóricos

BLOQUE 1: ASPECTOS AVANZADOS DE LA PROGRAMACIÓN

*Tema 1. Punteros
*Tema 2. Estructura de un programa en tiempo de ejecución. Memoria dinámica
*Tema 3. Ficheros: De texto y binarios
*Tema 4. Recursividad
*Tema 5. Estructuras lineales dinámicas de datos: Listas, pilas y colas
*Tema 6. Algoritmos básicos de búsqueda y ordenación y su complejidad algorítmica

BLOQUE II: ASPECTOS METODOLÓGICOS DE LA PROGRAMACIÓN

* Tema 7. Documentación y pruebas
* Tema 8. Herramientas (generación automática de proyectos, documentación, bibliotecas, depuradores, etc.)

2. Contenidos prácticos

*Familiarizar al alumno con algún entorno de depuración de programas.
*Aplicar el concepto de puntero.
*Gestionar de manera correcta la memoria dinámica.
*Aplicar los conceptos de ficheros a programas desarrollados en C.
*Aplicar el concepto de recursividad a la práctica.
*Familiarización con el uso de listas simplemente enlazadas.
*Programar algoritmos de ordenación.
*Utilizar el depurador y las bibliotecas.
*Familiarización con el uso de herramientas de generación automática de proyectos y de generación automática de documentación.

OBJETIVOS DE DESARROLLO SOSTENIBLE RELACIONADOS CON LOS CONTENIDOS

Sin relación

UNIVERSIDAD DE CÓRDOBA
www.uco.es
facebook.com/universidadcordoba
@univcordoba
INFORMACIÓN SOBRE TITULACIONES DE LA UNIVERSIDAD DE CÓRDOBA
uco.es/grados

GUÍA DOCENTE

METODOLOGÍA

Aclaraciones generales sobre la metodología (opcional)

La metodología docente propuesta considera el porcentaje de presencialidad de la asignatura indicado en el plan de estudios, según el cual la asignatura contempla 60 horas de trabajo presencial y 90 horas de trabajo no presencial. Por tanto, para conseguir un adecuado aprovechamiento del curso, adquirir las competencias y superar la asignatura, se debe asistir a clase y realizar regularmente trabajo personal en casa.

Lecciones magistrales

En estas sesiones se presentarán los conceptos de cada tema a la par que se desarrollarán ejemplos que ayuden al alumno a comprender y clarificar los conceptos. Los ejemplos consistirán en la resolución de problemas de creciente dificultad a lo largo del curso mediante el desarrollo de algoritmos y su posterior implementación.
Se impartirán en un aula de teoría con la pizarra y el cañón como medios didácticos fundamentales y con el apoyo del ordenador, con el objeto de poder compilar y ejecutar los ejemplos. La labor del alumno en estas clases consistirá en:
*Trasladar a sus apuntes las principales ideas que el profesor transmita y preguntar las dudas que le puedan surgir.
*Participar en la resolución de los ejercicios y problemas propuestos por el profesor.
Una vez en su casa, el alumno debería repasar la clase, comprendiendo los conceptos teóricos y repasando los ejercicios, por si hubiera alguna duda o alternativa en su solución. Estas dudas podrán ser planteadas al profesor en la siguiente clase o más tranquilamente en su horario de tutorías. Para un mejor aprovechamiento de estas clases, es muy recomendable que el alumno venga a clase con el material de la asignatura para que tome anotaciones sobre éste.

Laboratorio

Las clases de prácticas se realizan en un aula de ordenadores con el software necesario para implementar los programas. El objetivo de estas clases consiste en la implementación en el ordenador de un conjunto de ejercicios de programación cuyo enunciado estará previamente a disposición del alumno en la página web de la asignatura. Al comenzar la sesión el profesor comentará los enunciados de los ejercicios propuestos y, si procede, explicará alguna herramienta (bibliotecas, etc.). El resto de la clase se dedicará a que el alumno resuelva individualmente los ejercicios consultando con el profesor aquellas dudas que le surjan. El alumno debería terminar en casa aquellos ejercicios que no le haya dado tiempo a terminar. Para un correcto aprovechamiento de estas clases, se recomienda que el alumno previamente en casa haya estudiado los conceptos que se practicarán en la sesión de laboratorio y haya analizado el enunciado de los ejercicios propuestos.

Tutorías

Durante el desarrollo de la asignatura, cada alumno dispondrá, además del foro que se colocará en Moodle para la exposición y resolución de dudas de forma colectiva, de las tutorías individualizadas que desee dentro del horario establecido por el profesor. Igualmente, se han programado sesiones de tutorías colectivas que se celebrarán a lo largo del curso.

Resolución de problemas

Se usarán, como apoyo a las clases teóricas y prácticas, y se realizarán de forma individual por cada alumno en el aula. Estas sesiones se centrarán en el análisis de problemas y su planteamiento como algoritmos o programas.

Adaptaciones metodológicas para alumnado a tiempo parcial y estudiantes con discapacidad y necesidades educativas especiales

*Los alumnos a tiempo parcial seguirán la misma metodología de los alumnos a tiempo completo. Se podrán realizar adaptaciones en función de cada caso y previo acuerdo entre la profesora coordinadora y el alumno a principio de curso.

GUÍA DOCENTE

*En el caso de los alumnos con discapacidad y necesidades educativas especiales, se llevarán a cabo adaptaciones siguiendo las pautas marcadas por los órganos competentes (Escuela, Unidad de Educación Inclusiva, Gabinete de Atención Psicológica, etc.).

Actividades presenciales

Actividad	Grupo completo	Grupo mediano	Total
Actividades de evaluación	2	2	4
Laboratorio	-	22	22
Lección magistral	26	-	26
Resolución de problemas	6	-	6
Tutorías	2	-	2
Total horas:	***36***	***24***	***60***

Actividades no presenciales

Actividad	Total
Consultas bibliográficas	10
Ejercicios	49
Estudio	31
Total horas:	***90***

MATERIAL DE TRABAJO PARA EL ALUMNO

Cuaderno de Prácticas - *http://www.uco.es/moodle*
Ejercicios y problemas - *http://www.uco.es/moodle*
Presentaciones PowerPoint - *http://www.uco.es/moodle*

Aclaraciones

*Presentaciones PowerPoint. El alumno recibirá a lo largo del curso las presentaciones PowerPoint que se van a explicar clase. Este material se debe estudiar, asimilar, y completar con el estudio de referencias bibliográficas, la realización de ejercicios y la búsqueda de información.
*Cuaderno de prácticas. El alumno recibirá a lo largo del curso una serie de guiones de prácticas con ejercicios propuestos que deberá resolver en ordenador.
*Ejercicios y problemas. El alumno tendrá a su disposición en Moodle ejercicios resueltos que serán utilizados en las clases de tutorías colectivas.

GUÍA DOCENTE

EVALUACIÓN

Competencias	Exámenes	Resolución de problemas	Supuesto práctico/discusión caso clínico/discusión trabajo científico
CB4		X	
CEB4	X		X
CEB5	X		X
CU2	X		X
Total (100%)	***70%***	***10%***	***20%***
Nota mínima (*)	***4***	***0***	***0***

(*)Nota mínima (sobre 10) necesaria para que el método de evaluación sea considerado en la calificación final de la asignatura. En todo caso, la calificación final para aprobar la asignatura debe ser igual o superior a 5,0.

Valora la asistencia en la calificación final:

No

Aclaraciones generales sobre los instrumentos de evaluación:

Siguiendo lo establecido en el Reglamento de Régimen Académico (artículo 28) y las recomendaciones del informe de reacreditación del Grado (página 7), se ha establecido la siguiente correspondencia entre los instrumentos disponibles en eguiado y los sistemas de evaluación de y pesos específicos del módulo de formación básica establecidos en la memoria verificada:

*Pruebas de micros (20%) -> Casos y supuestos prácticos (bloque 1 verifica) -> Supuesto práctico (eguiado)
*Cuestionarios (10%) -> Autoevaluación (bloque 3 verifica) -> Resolución de problemas (eguiado)
*Examen práctico (50%) -> Respuesta larga (bloque 2 verifica) -> Exámenes (eguiado)
*Examen teórico (20%) -> Pruebas objetivas (bloque 2 verifica) -> Exámenes (eguiado)

Se recuerda que el Reglamento de Régimen Académico prohíbe acceder al recinto del examen portando dispositivos electrónicos que permitan la comunicación o el almacenamiento de datos.

EVALUACIÓN EN LA PRIMERA CONVOCATORIA ORDINARIA DEL SEGUNDO CUATRIMESTRE

***Pruebas de micros (20%)**. Durante el cuatrimestre se realizarán dos pruebas (o micros-exámenes) en clase de prácticas evaluados de 0 a 10. Solo se prodrán realizar en el día fijado a tal efecto. Cada una de las pruebas consistirá en resolver un ejercicio de programación, delante del ordenador, referente a los contenidos vistos en la asignatura hasta el momento. Esta calificación se guarda solo hasta la segunda convocatoria ordinaria para calcular la Nota1.

***Cuestionarios (10%)**. A lo largo del cuatrimestre se propondrán una serie de cuestionarios sobre los temas impartidos. Solo se podrán realizar durante el periodo de clases dentro del plazo establecido. Esta calificación se guarda solo hasta la segunda convocatoria ordinaria para el cálculo de la Nota1.

***Examen práctico (50%)**. Examen sobre el ordenador en el que el alumno deberá realizar diferentes programas en un tiempo determinado, a cuyo término deberán compilar y funcionar correctamente. El alumno deberá demostrar que se desenvuelve bien en el desarrollo de programas y de las herramientas: editores, compiladores, depuradores, bibliotecas, etc. Podrá incluir un problema de nivel medio eliminatorio, de forma que sólo se

GUÍA DOCENTE

corregirá el examen del resto de la prueba si el alumno lo supera. Si el alumno ha superado las pruebas de micros (media_micros >=5 sobre 10), dicho problema no tendrá el carácter de eliminatorio. Esta prueba se evalúa de 0 a 10 y deben completarse todos los ejercicios (incluido el de carácter eliminatorio) para optar a la máxima puntuación.

***Examen teórico (20%)**. Prueba teórica, evaluada de 0 a 10, en la que el alumno debe demostrar sus conocimientos contestando a una serie de preguntas sobre los contenidos teórico/prácticos vistos durante el curso. En la primera convocatoria, el alumno deberá realizar obligatoriamente los exámenes teórico y práctico.

Para considerar que es correcta la resolución de los ejercicios propuestos en las pruebas de micros y en el exámen práctico, deberán compilar y solucionar el problema propuesto sin fallos de ejecución para cualquier entrada de datos válida, aunque sea diferente de los datos de ejemplo proporcionados.

Para ponderar y sumar todas las partes, hay que obtener al menos un 4 sobre 10 en el examen teórico y un 4 sobre 10 en el examen práctico, en caso contrario la calificación final será suspenso. Para aprobar, la nota final después de ponderar y sumar todas las partes debe ser superior o igual a 5 sobre 10.

EVALUACIÓN EN LA SEGUNDA CONVOCATORIA ORDINARIA DEL SEGUNDO CUATRIMESTRE

En la segunda convocatoria, el alumno deberá realizar obligatoriamente los exámenes teórico y práctico. El examen práctico podrá contener un problema eliminatorio. Una vez realizados los exámenes, se calcularán dos notas, siendo la nota final la más favorable para el alumno:

***Nota1**: 10% cuestionarios, 20% micros, 50% examen práctico de la segunda convocatoria, 20% examen teórico de la segunda convocatoria.

***Nota2**: 30% examen teórico de la segunda convocatoria y 70% examen práctico de la segunda.

Para considerar que es correcta la resolución de los ejercicios propuestos en las pruebas de micros y en el exámen práctico, deberán compilar y solucionar el problema propuesto sin fallos de ejecución para cualquier entrada de datos válida, aunque sea diferente de los datos de ejemplo proporcionados.

Tanto para la Nota1 como para la Nota2, para ponderar y sumar todas las partes hay que obtener al menos un 4 sobre 10 en el examen teórico y un 4 sobre 10 en el examen práctico, en caso contrario la calificación final será suspenso. Para aprobar, la nota final después de ponderar y sumar todas las partes debe ser superior o igual a 5 sobre 10.

Aclaraciones sobre la evaluación para el alumnado a tiempo parcial y necesidades educativas especiales:

*Los alumnos a tiempo parcial seguirán la misma metodología de los alumnos a tiempo completo. Se podrán realizar adaptaciones en función de cada caso y previo acuerdo entre la profesora coordinadora y el alumno a principio de curso.

*En el caso de los alumnos con discapacidad y necesidades educativas especiales, se llevarán a cabo adaptaciones siguiendo las pautas marcadas por los órganos competentes (Escuela, Unidad de Educación Inclusiva, Gabinete de Atención Psicológica, etc.).

Aclaraciones sobre la evaluación de la convocatoria extraordinaria y convocatoria extraordinaria de finalización de estudios:

EVALUACIÓN EN LA CONVOCATORIA EXTRAORDINARIA

Al tratarse de una convocatoria extraordinaria, solamente se podrán presentar aquellos alumnos que cumplan los requisitos establecidos por la Universidad.

Se realizarán obligatoriamente el examen teórico y el práctico. Los criterios serán 30% examen teórico y 70%

GUÍA DOCENTE

examen práctico que podrá contener ejercicio eliminatorio. Para ponderar y sumar las partes, hay que obtener al menos un 4 sobre 10 en el examen teórico y un 4 sobre 10 en el examen práctico. Para aprobar, la nota final después de ponderar y sumar todas las partes debe ser superior o igual a 5 sobre 10.

EVALUACIÓN EN LA CONVOCATORIA EXTRAORDINARIA DE FIN DE ESTUDIOS

Al tratarse de una convocatoria extraordinaria, solamente se podrán presentar aquellos alumnos que cumplan los requisitos establecidos por la Universidad.

Se realizarán obligatoriamente el examen teórico y el práctico. Los criterios serán 30% examen teórico y 70% examen práctico que podrá contener ejercicio eliminatorio. Para ponderar y sumar las partes, hay que obtener al menos un 4 sobre 10 en el examen teórico y un 4 sobre 10 en el examen práctico. Para aprobar, la nota final después de ponderar y sumar todas la partes debe ser superior o igual a 5 sobre 10.

Criterios de calificación para la obtención de Matrícula de Honor:

La Matrícula de Honor podrá ser otorgada a estudiantes cuya calificación final sea igual o superior a 9 que hayan mostrado un desempeño excelente en la asignatura, y siempre según lo establecido en el Reglamento de Régimen Académico (artículo 80.3).

BIBLIOGRAFIA

1. Bibliografía básica

Bibliografía básica

*Tema 6. Bratley, P. and Brassard, G. Fundamentos de Algoritmia. Pearson Educación, 2004. capítulos 2, 3 y 4.

*Tema 1, Tema 2, Tema 3, Tema 4, Tema 5, Tema 6, Tema 7. Joyanes, L. and Zahonero, I. Programación en C: metodología, algoritmos y estructuras de datos. McGraw-Hill, 2005. Capítulos 2, 8, 10, 12, 13, 14, 15.

*Tema 1. Gottfried, B. and Lázaro, J. Programación en C. McGraw-Hill Interamericana de España S.L., 2005. Capítulo 10.

*Tema 2, Tema 8. Jones, B.; Aitken, P. and Miller, D. Sams. Teach Yourself C Programming in One Hour a Day. Sams, 2013. Capitulos 21, 22.

Bibliografía avanzada

*Tema 1, Tema 3. Deitel, P. and Deitel, H. C for Programmers with an Introduction to C11. Pearson Education, 2013. Capitulos 7 y 11.

*Tema 1, Tema 2, Tema 8. Prinz, P. and Crawford, T. C in a Nutshell: The Definitive Reference. O'Reilly Media, 2015. Capítulos 9, 12, 20, 21.

Bibliografía para ejercicios

*Joyanes, L. Problemas de metodología de la programación. McGraw-Hill, 1990. *Joyanes, L. Programación en C: Libro de problemas. McGraw-Hill, 2002.

2. Bibliografía complementaria

*Perry, G. and Miller, D. C. Programming Absolute Beginner's Guide. Pearson Education, 2013.

*Kernighan, B. and Ritchie, D. El lenguaje de programación C. Pearson Educación, 1991.

*Deitel, H.; Deitel, P. and Garcia, G. Cómo programar en C/C++. Pearson Educación, 1995.

GUÍA DOCENTE

CRITERIOS DE COORDINACIÓN

Fecha de entrega de trabajos

Aclaraciones

Se hará uso de las herramientas proporcionadas desde la Coordinación del Grado para planificar las fechas en que tendrán lugar los micros y las recuperaciones de clase si es que fuera necesario.

Las estrategias metodológicas y el sistema de evaluación contempladas en esta Guía Docente serán adaptadas de acuerdo a las necesidades presentadas por estudiantes con discapacidad y necesidades educativas especiales en los casos que se requieran.