



UNIVERSIDAD DE LAS PALMAS
DE GRAN CANARIA

GUÍA DOCENTE

CURSO: 2011/12

40803 - INTRODUCCIÓN A LA INFORMÁTICA

CENTRO: 180 - Escuela de Ingeniería Informática

TITULACIÓN: 4008 - Grado en Ingeniería Informática

ASIGNATURA: 40803 - INTRODUCCIÓN A LA INFORMÁTICA

Vinculado a : (Titulación - Asignatura - Especialidad)

4801-Doble Grado en Ingeniería Informática y - 48102-INTRODUCCIÓN A LA INFORMÁTICA - 00

CÓDIGO UNESCO: 1203

TIPO: Básica de Rama

CURSO: 1

SEMESTRE: 1º semestre

CRÉDITOS ECTS: 6

Especificar créditos de cada lengua:

ESPAÑOL: 6

INGLÉS: 0

SUMMARY

REQUISITOS PREVIOS

Los propios de un alumno que accede al primer curso del Grado en Ingeniería Informática.

Datos identificativos del profesorado que la imparte.

Plan de Enseñanza (Plan de trabajo del profesorado)

Contribución de la asignatura al perfil profesional:

Introducción a la Informática presenta los conceptos básicos de la programación y pone en contacto al estudiante con los entresijos de los computadores. Por un lado, el estudiante accederá al mundo de la programación de forma progresiva, enfrentándose a los primeros problemas algorítmicos que le irán dando soltura a la hora de seleccionar los esquemas más adecuados a cada problema. Por otro lado, el estudiante tomará contacto con la estructura interna de un computador y su funcionamiento, aprendiendo a operar con diferentes sistemas de numeración.

Competencias que tiene asignadas:

Competencias generales: G1, G2, G3, G4, G5

Competencias de la universidad: N1, N2, N3, N4, N5

Competencias del título: T1, T5, T6, T8, T9, T10

Competencias del módulo de formación básica: FB04, FB05

Objetivos:

El principal objetivo de esta asignatura es introducir los conocimientos básicos que el estudiante necesitará para afrontar las asignaturas posteriores tanto del área de Lenguajes y Sistemas Informáticos como del área de Arquitectura y Tecnología de Computadores.

Contenidos:

CONTENIDOS TEÓRICOS

Módulo I: Introducción a la Programación

- 1.1 Software, clasificación, software de sistemas y aplicaciones
- 1.2 Algoritmos
- 1.3 Constantes, variables, expresiones y entrada/salida
- 1.4 Tipos de datos
- 1.5 Subprogramas
- 1.6 Tipos de datos compuestos

Módulo II: Introducción a los Computadores

- 2.1 Esquema de funcionamiento de un computador
- 2.2 Representación de la información
- 2.3 Lenguajes máquina y ensamblador
- 2.4 Sistemas de numeración, operaciones y códigos

CONTENIDOS PRÁCTICOS

Módulo I: Introducción a la Programación

- 1.1 Entorno de desarrollo
- 1.2 Estructuras básicas de control
- 1.3 Manejo de tipos de datos.
- 1.4 Subprogramas
- 1.5 Estructuras compuestas

Módulo II: Introducción a los Computadores

- 2.1 El simulador PCspim
- 2.2 Los datos en memoria, operaciones de carga y almacenamiento
- 2.3 Operaciones aritméticas y lógicas
- 2.4 Estructuras de control
- 2.5 Gestión de subprogramas

Metodología:

- Sesiones académicas teóricas.
- Sesiones académicas prácticas.
- Sesiones académicas de problemas.
- Trabajo personal y resolución de ejercicios.
- Lecturas obligatorias.
- Tutorías colectivas.
- Tutorías individuales.

Criterios y fuentes para la evaluación:

- Exámenes.
- Controles de asistencia a las sesiones académicas.
- Controles de las actividades académicamente dirigidas.
- Evaluación continua de los contenidos teóricos y prácticos.
- Controles de los ejercicios realizados.
- Controles de lecturas obligatorias.
- Participación activa en las sesiones académicas.

- Participación activa en las tutorías.

Sistemas de evaluación:

La asignatura se divide en dos módulos que se evaluarán por separado: Introducción a la Programación e Introducción a los Computadores. A su vez, cada módulo se divide en otras dos partes que también se evaluarán por separado: teoría y prácticas.

Durante la convocatoria ordinaria, la parte teórica de Introducción a los Computadores podrá ser evaluada de dos formas: evaluación continua o evaluación no continua. En el resto de convocatorias se evaluará de forma no continua.

Criterios de calificación:

La calificación final de la asignatura se calculará en función de la calificación correspondiente a cada una de sus cuatro partes:

- Teoría de Introducción a la Programación: 25% de la nota final
- Prácticas de Introducción a la Programación: 25% de la nota final
- Teoría de Introducción a los Computadores: 25% de la nota final
- Prácticas de Introducción a los Computadores: 25% de la nota final

Es necesario aprobar cada una de las cuatro partes por separado, obteniendo una nota igual o superior a cinco, para aprobar la asignatura. En caso de no aprobar alguna de las partes, la calificación del estudiante será un suspenso con una nota máxima en acta de 4.

La evaluación no continua de la parte teórica de Introducción a los Computadores se realizará por medio de un examen en cada convocatoria. La evaluación continua de esta parte durante la convocatoria ordinaria se realizará en función de una serie de actividades:

- Lecturas obligatorias. Durante el curso se realizará un test por cada una de las lecturas obligatorias. Las notas de estos test contribuirán con un 20% a la nota final, teniendo todos ellos igual peso.
- Sesiones de ejercicios. Durante el curso se realizará una serie de ejercicios. Las notas de estos ejercicios contribuirán con un 20% a la nota final, teniendo todos ellos igual peso.
- Pruebas escritas. Durante el curso se realizará una serie de pruebas escritas de carácter no liberatorio. Las notas de estas pruebas contribuirán con un 60% a la nota final, teniendo todas ellas igual peso.

No se exigirá nota mínima en ninguna de las actividades para aprobar esta parte de la asignatura. Sin embargo, para acogerse a la evaluación continua, los estudiantes deben cumplir una serie de requisitos:

- Asistir al menos al 80% de las sesiones de teoría y al 80% de las sesiones de laboratorio.
- Entregar todos los ejercicios, realizar todos los test de lecturas obligatorias y realizar todas las pruebas escritas.

Los estudiantes que no cumplan estas condiciones no podrán acogerse a la evaluación continua independientemente de las actividades que hayan podido realizar.

Plan de Aprendizaje (Plan de trabajo de cada estudiante)

Tareas y actividades que realizará según distintos contextos profesionales (científico, profesional, institucional, social)

- Asistencia a clases magistrales y sesiones de resolución de problemas en el aula.
- Asistencia a sesiones prácticas en el laboratorio.
- Asistencia a tutorías individuales y/o colectivas.
- Sesiones de estudio, sesiones de ejercicios, sesiones de lectura y trabajo personal.

- Resolución de ejercicios sobre algorítmica y programación en lenguajes de alto nivel.
- Resolución de ejercicios sobre representación de información, sistemas de numeración, operaciones y códigos.
- Resolución de ejercicios sobre programación en lenguaje máquina y ensamblador.

Temporalización semanal de tareas y actividades (distribución de tiempos en distintas actividades y en presencialidad - no presencialidad)

Actividades presenciales: 4 horas a la semana.

- Teoría (T): 2 horas a la semana.
- Prácticas (P): 2 horas a la semana.
- Exámenes: se realizarán durante horas de teoría.

Actividades no presenciales: 6 horas a la semana.

- Sesiones de trabajo personal (TP).
- Lecturas obligatorias (LO).
- Ejercicios (E).

Semana 1: 2T + 2P + 6TP

Semana 2: 2T + 2P + 6TP

Semana 3: 2T + 2P + 6TP

Semana 4: 2T + 2P + 6TP

Semana 5: 2T + 2P + 6TP

Semana 6: 2T + 2P + 6TP

Semana 7: 2T + 2P + 6TP

Semana 8: 2T + 2P + 6TP

Semana 9: 2T + 2P + 2LO + 4TP

Semana 10: 2T + 2P + 2LO + 2E + 2TP

Semana 11: 2T + 2P + 2LO + 2E + 2TP

Semana 12: 2T + 2P + 2LO + 2E + 2TP

Semana 13: 2T + 2P + 2LO + 2E + 2TP

Semana 14: 2T + 2P + 2LO + 2E + 2TP

Semana 15: 2T + 2P + 2E + 4TP

Recursos que tendrá que utilizar adecuadamente en cada uno de los contextos profesionales.

- Pizarra y transparencias.
- Ordenador personal.
- Simulador PCspim.
- GNAT Programming Studio.
- Plataforma Moodle (Campus Virtual ULPGC).

Resultados de aprendizaje que tendrá que alcanzar al finalizar las distintas tareas.

Al finalizar las tareas correspondientes al módulo de Introducción a la Programación, un estudiante deberá ser capaz de:

- Describir, diferenciar y relacionar los elementos fundamentales de la programación estructurada y saber usarlos de forma adecuada en el desarrollo de programas.
- Crear algoritmos para resolver problemas simples, y aplicar la metodología de análisis descendente para descomponer los problemas de mediana complejidad en subproblemas más simples.
- Explicar el concepto de tipo de dato y los tipos de datos básicos, y definir y usar tipos de datos

adecuados en la resolución de problemas simples.

- Seleccionar los esquemas algorítmicos básicos de recorrido y búsqueda y aplicarlos de forma adecuada para resolver problemas concretos.
- Justificar la utilidad de los elementos de estilo y usarlos de forma adecuada en el desarrollo de programas.

Al finalizar las tareas correspondientes al módulo de Introducción a los Computadores, un estudiante deberá ser capaz de:

- Describir la estructura interna y el funcionamiento de un computador.
- Representar datos numéricos y no numéricos.
- Operar en los sistemas de numeración binario, octal y hexadecimal.
- Realizar programas sencillos en lenguaje ensamblador.

Plan Tutorial

Atención presencial individualizada (incluir las acciones dirigidas a estudiantes en 5ª, 6ª y 7ª convocatoria)

Todos los profesores de la asignatura dedican seis horas semanales a la atención presencial. El horario de atención presencial de cada profesor está publicado en la página web del Departamento de Informática y Sistemas.

Atención presencial a grupos de trabajo

Todos los profesores de la asignatura dedican seis horas semanales a la atención presencial. El horario de atención presencial de cada profesor está publicado en la página web del Departamento de Informática y Sistemas.

Atención telefónica

Todos los profesores de la asignatura están disponibles para atención telefónica durante los horarios de atención presencial. El número de teléfono de cada profesor está publicado en la página web del Departamento de Informática y Sistemas.

Atención virtual (on-line)

Todos los profesores de la asignatura están disponibles para atender a los estudiantes virtualmente por medio de su dirección de correo electrónico pero preferentemente a través de las herramientas proporcionadas por el Campus Virtual de la universidad. La dirección de correo electrónico de cada profesor está publicada en la página web del Departamento de Informática y Sistemas.

Bibliografía

[1 Básico] Principios de diseño digital /

Daniel D. Gajski.

Prentice Hall,, Madrid : (2000)

8483220040

[2 Básico] Estructura y diseño de computadores: interficie circuitería, programación

David A. Patterson, John L. Hennessy.

Reverté,, Barcelona : (2004) - (reimp.)

84-291-2618-X(v.3)

[3 Básico] Programming in ADA 95 /

John Barnes.

Addison-Wesley,, Harlow (England) : (1998) - (2nd ed.)

0-201-34293-6

[4 Básico] Fundamentos de sistemas digitales /

Thomas L. Floyd.

Prentice Hall,, Madrid : (2000) - (7ª ed.)

84-205-2994-X

[5 Básico] Organización y arquitectura de computadores /

William Stallings.

Prentice Hall,, Madrid : (2006) - (7ª ed.)

8489660824

[6 Básico] Iniciación a la programación Ada 2005 como primer lenguaje /

Zenón

J. Hernández Figueroa, Francisco J. Carreras Riudavets, Gustavo Rodríguez Rodríguez, José Daniel González Domínguez.

Lulu.com,, [s.l.] : (2009) - (2ª ed.)

9781409268000

[7 Básico] Fundamentos de estructuras de datos: soluciones en Ada, Java y C++ /

Zenón José Hernández Figueroa, Juan Carlos Rodríguez del Pino, José Daniel González Domínguez, Margarita Díaz Roca, José Rafael Pérez Aguilar, Gustavo Rodríguez Rodríguez ... [et al.].

Thomson,, Australia, España [etc.] : (2005)

84-9732-358-0

[8 Recomendado] Una introducción a la programación: un enfoque algorítmico /

Jesús J. García Molina, ...[et al.].

Thomson,, Madrid : (2005)

8497321855

[9 Recomendado] Fundamentos de programación: algoritmos y estructura de datos /

Luis Joyanes Aguilar.

McGraw-Hill,, Madrid : (1988)

8476152019
