ASIGNATURA / COURSE TITLE

LABORATORIO / LABORATORY

1.1. Código / Course number

16440

1.2. Materia/ Content area

EL LENGUAJE MATEMÁTICO

1.3. Tipo / Course type

Formación básica / Compulsory subject

1.4. Nivel / Course level

Grado / Bachelor (first cycle)

1.5. Curso / Year

1º / 1st

1.6. Semestre / Semester

Anual / Annual

1.7. Idioma / Language

Español. Se emplea también Inglés en material docente / In addition to Spanish, English is also extensively used in teaching material

1.8. Requisitos previos / Prerequisites

Ninguno específico / None

1.9. Requisitos mínimos de asistencia a las sesiones presenciales/ Minimun attendance requirement

La asistencia a clase es obligatoria. El estudiante deberá justificar las faltas de asistencia. De cualquier forma, el estudiante es responsable de conocer toda la información que se transmita en las clases presenciales.

1.10. Datos del equipo docente / Faculty data

Rafael Hernández García
Teléfono: 91 497-4258
Despacho: Módulo 17 – Desp. 512
Correo Electrónico: rafael.hernandez@uam.es
Página web:
Horario de atención: a discreción

El resto del profesorado implicado en la asignatura puede consultarse en la página web del título:

http://www.uam.es/ss/Satellite/Ciencias/es/1242671471248/listadoCombo/Profesorado.htm

1.11. Objetivos del curso / Course objectives

Tener (si no se ha tenido antes) un primer contacto con las ideas y estructuras básicas de programación, como los bucles y las subrutinas, así como enunciar y exponer con claridad el método de resolución utilizado en cada problema, los resultados positivos obtenidos y, en su caso, las carencias.

Los Resultados del Aprendizaje evaluables que el alumno habrá alcanzado al superar esta asignatura son:

R1.15· Conocerá, entenderá y sabrá utilizar herramientas informáticas como apoyo a la resolución de algunos problemas matemáticos sencillos.
R1.16· Será capaz de enfrentar un problema e indagar en la herramienta informática elegida, sabiendo discernir qué parte del problema es posible resolver y cuál necesita herramientas más sofisticadas, o una teoría más desarrollada.
R1.17· Manejará el lenguaje de la programación.

R1.18· Sabrá aplicar cálculo numérico para la solución de problemas planteados.
R1.19. Será capaz de realizar un análisis razonado a partir de gráficos de datos y de esbozos de gráficas de funciones.

Estos Resultados del Aprendizaje contribuyen a la adquisición de las siguientes Competencias Generales y Específicas del título:

COMPETENCIAS GENERALES	
G1	Conocer los conceptos, métodos y resultados más relevantes de las diferentes ramas de las matemáticas.
G2	Aplicar tanto los conocimientos como la capacidad de análisis y de abstracción adquiridos en la definición y planteamiento de problemas y en la búsqueda de sus soluciones tanto en contextos académicos como profesionales.
G3	Recabar e interpretar datos, información o resultados relevantes en problemas científicos, tecnológicos o de otros ámbitos que requieran el uso de herramientas matemáticas. Obtener conclusiones y exponerlas razonadamente.
G7	Utilizar herramientas de búsqueda de recursos bibliográficos en Matemáticas.

COMPETENCIAS ESPECÍFICAS	
E1	Comprender y utilizar el lenguaje matemático. Adquirir la capacidad para enunciar proposiciones en distintos campos de la Matemática, para construir demostraciones y para transmitir los conocimientos matemáticos adquiridos.
E2	Conocer demostraciones rigurosas de algunos teoremas clásicos en distintas áreas de la Matemática.
E3	Asimilar la definición de un nuevo objeto matemático, en términos de otros ya conocidos, y ser capaz de utilizar este objeto en diferentes contextos.
E4	Saber abstraer las propiedades estructurales (de objetos matemáticos, de la realidad observada, y de otros ámbitos) distinguiéndolas de aquellas puramente ocasionales y poder comprobarlas con demostraciones o refutarlas con contraejemplos, así como identificar errores en razonamientos incorrectos.
E5	Proponer, analizar, validar e interpretar modelos de situaciones reales sencillas, utilizando las herramientas matemáticas más adecuadas a los fines que se persigan.
E6	Resolver problemas de Matemáticas, mediante habilidades de cálculo básico y otros, planificando su resolución en función de las herramientas de que se disponga y de las restricciones de tiempo y recursos.
E7	Utilizar aplicaciones informáticas de análisis estadístico, cálculo numérico y simbólico, visualización gráfica, optimización u otras para experimentar en Matemáticas y resolver problemas.

1.12. Contenidos del programa / Course contents

A diferencia de las otras asignaturas, destinada cada una a presentar determinadas teorías y conceptos, el Laboratorio pretende

1. servir de apoyo a las demás en aquellos asuntos que se enriquezcan y/o se hagan viables con ayuda de la máquina;

2. transmitir, al hacerlo, las destrezas básicas de uso del software, y cierta visión unificada o transversal de ideas y objetos que aparecen en distintos programas bajo diferente punto de vista.

Por ello, no consta de Capítulos que deban irse cubriendo uno tras otro, sino de unos cuantos 'Temas' que irán apareciendo desde el principio y reapareciendo de manera flexible al calor de los mencionados asuntos de las otras asignaturas; y que en ciertos momentos se verán fundidos en alguna idea común a varios de esos Temas.

Tema 1. Uso de *SAGE* como una "calculadora científica": Manipulación de matrices, resolución de problemas de álgebra lineal, cálculo diferencial e integral. Representaciones gráficas.
Tema 2. Aspectos básicos de programación en *Python*: estructuras de datos, funciones, control de flujo, recursión.
Tema 3. Ejemplos en teoría elemental de números, números primos.
Tema 4. Aproximación de números reales.
Tema 5. Resolución de problemas "por fuerza bruta", combinatoria.
Tema 6. Criptografía elemental.
Tema 7. Probabilidad y simulación usando generadores de números (pseudo) aleatorios.
Tema 8. Miscelánea.

1.13. Referencias de consulta / Course bibliography

SAGE

SAGE Tutorial
http://www.sagemath.org/doc/tutorial/

Ayuda sobre SAGE
http://sagemath.org/help.html

Manual de SAGE para principiantes
http://sagemath.org/es/Manual_SAGE_principiantes.pdf
T. Kosan

Introducción a la programación en python

An introduction to Python

Rossum, Guido van
Network Theory, 2006

Non-Programmer's Tutorial for Python 2.6
wikibook: varios autores
http://en.wikibooks.org/wiki/Non-Programmer%27s_Tutorial_for_Python_2.6

A byte of Python
Swaroop C H
http://www.swaroopch.com/notes/Python

Manuales avanzados de python

How to think like a computer scientist:Learning with python
Allen Downey, Jeffrey Elkner, Chris Meyers

A primer on scientific programming with python
Hans Petter Langtangen

2. Métodos Docentes / Teaching methodology

Esta asignatura se organiza mediante clases presenciales de teoría y prácticas (2 horas semanales) a las que se añaden las horas de trabajo personal del estudiante para el estudio y la resolución de ejercicios o trabajos planteados por el profesor (al menos 2 horas semanales). Se usarán también parte de las clases presenciales para la realización de exámenes, controles intermedios u otras actividades.
En media semanal, las horas presenciales se distribuyen en:
1 hora de teoría y problemas (en las que se imparten los contenidos teóricos acompañados de ejercicios y ejemplos y se resuelven algunos de los problemas planteados a los estudiantes)
1 hora de prácticas (en las que se pretende una participación activa del estudiante a través de la resolución de ejercicios y problemas, presentaciones de trabajos, realización de controles intermedios, etc.)

El curso consta de las siguientes actividades: clases teórico-prácticas en el laboratorio de informática, tutorías y exámenes en el aula.

Las clases incluyen el planteamiento de un problema, la exposición de los posibles contenidos teóricos necesarios para su resolución, la discusión de ejemplos sencillos y el trabajo individual del alumno en un ordenador.

Se dispone de una página web en la que se cuelgan materiales de apoyo, ejemplos prácticos y ejercicios. Como sistema de apoyo a la docencia los estudiantes disponen de tutorías individuales y electrónicas.

3. Tiempo de trabajo del estudiante / Student workload

2 horas/semana: clases de práctica con ordenador
2 horas/semana: horas de trabajo independiente del estudiante
5 horas/curso: exámenes de evaluación presenciales

4. Métodos de evaluación y porcentaje en la calificación final / Evaluation procedures and weight of components in the final grade

La nota de curso (NC) será el resultado de varias evaluaciones, repartidas a lo largo del mismo.

La prueba final (calificación=EF) se realizará en la fecha publicada oficialmente, en un aula de informática. No podrá presentarse al examen final, en su convocatoria ordinaria, ningún alumno que no haya obtenido una calificación igual o superior a 3 en al menos una de las evaluaciones.

La convocatoria extraordinaria (calificación=Ex) consistirá en un examen en aula de ordenadores.

La calificación final del curso será

- la nota de curso, NC;
- conforme a la fórmula: 0.5 * EF + 0.5 * NC, si el alumno se presenta a la prueba final (convocatoria ordinaria);
- con la fórmula: 0.7*Ex+0.3*NC, si el alumno ha de presentarse a la convocatoria extraordinaria.

5. Cronograma* / Course calendar

Las clases se desarrollarán con una periodicidad semanal, a lo largo de todo el curso. Cada semana habrá una clase de dos horas, dividida en una primera parte teórica y una segunda parte práctica.

Los contenidos concretos de cada semana deberán adaptarse al ritmo con el que los estudiantes asimilen las herramientas básicas de programación, que estarán especialmente concentradas en las primeras sesiones. Una vez superada esta primera fase, se pasará a cubrir con más detalle los distintos temas especificados en el apartado 1.12, en coordinación con los profesores del resto de asignaturas, para adaptar el ritmo de las distintas prácticas al desarrollo de los programas de las asignaturas con las que se relacionan.