



Guía docente de la asignatura

Asignatura	FÍSICA COMPUTACIONAL		
Materia	FÍSICA COMPUTACIONAL		
Módulo			
Titulación	PROGRAMA DE ESTUDIOS CONJUNTO DE GRADO EN FÍSICA Y GRADO EN MATEMATICAS		
Plan	469	Código	45742
Periodo de impartición	ANUAL	Tipo/Carácter	FORMACIÓN BÁSICA
Nivel/Ciclo	GRADO	Curso	1º
Créditos ECTS	6 ECTS		
Lengua en que se imparte	CASTELLANO		
Profesor/es responsable/s	LUIS ENRIQUE GONZÁLEZ TESEDO, JOSÉ MANUEL LÓPEZ RODRÍGUEZ, LUIS MIGUEL MOLINA MARTÍN.		
Datos de contacto (E-mail, teléfono...)	Luis Enrique González Tesedo , 983 183599 luisen@liq1.fam.cie.uva.es Despacho B115 José Manuel López Rodríguez , 983 423143 jmlopez@fta.uva.es Despacho B116, Luis Miguel Molina Martín , 983 185062 lmolina@fta.uva.es Despacho B123		
Horario de tutorías	http://www.uva.es/ -> Grados -> Grado en Física -> Tutorías Lunes-Viernes 16:00-17:00, Facultad de Ciencias, Primer piso, Bloque B (de color azul) y en cualquier otro día y horario, previa cita concertada por correo electrónico, en el mismo lugar.		
Departamento	Departamento de Física Teórica, Atómica y Óptica		



1. Situación / Sentido de la Asignatura

1.1 Contextualización

Se imparte dos horas a la semana a lo largo de todo el curso académico, en el primer curso del Grado de Física. Es una asignatura de formación básica. Esta asignatura consiste en programar en lenguaje C la resolución de problemas de Física y por tanto, es un complemento de las asignaturas teóricas del Grado de Física que imparten conocimientos de Matemáticas y Física.

1.2 Relación con otras materias

Es una de las dos asignaturas de la materia Física Computacional. Los contenidos de esta asignatura están relacionados con los contenidos de varias asignaturas del primer curso del Grado de Física: Fundamentos de Mecánica y Termodinámica, Fundamentos de Campos y Ondas, Fundamentos de Física Cuántica y Estadística, Análisis Matemático y Álgebra lineal y Geometría.

1.3 Prerrequisitos

Según la ficha de la asignatura los requisitos de esta asignatura son los generales de cualquier asignatura del Grado de Físicas: Conocimientos de física, álgebra, trigonometría y análisis matemático, a nivel de 1º y 2º de Bachillerato. Sin embargo, dos tercios de las clases presenciales consistirán en prácticas de programación en un ordenador del aula de informática y para ello es imprescindible saber manejar ordenadores y Windows a nivel de usuario y saber escribir a máquina a nivel de usuario.



2. Competencias

2.1 Generales y transversales

- T1: Capacidad de análisis y de síntesis.
- T2: Capacidad de organización y planificación.
- T3: Capacidad de comunicación oral y escrita.
- T4: Capacidad de resolución de problemas.
- T5: Capacidad de trabajar en equipo.
- T7: Capacidad de trabajo y aprendizaje autónomo.
- T8: Capacidad de adaptación a nuevas situaciones.
- T9: Creatividad.

2.2 Específicas

- E3: Ser capaz de comparar nuevos datos experimentales con modelos disponibles para revisar su validez y sugerir cambios con el objeto de mejorar la concordancia de los modelos con los datos.
- E6: Ser capaz de realizar las aproximaciones requeridas con el objeto de reducir un problema hasta un nivel manejable.
- E7: Ser capaz de desarrollar software propio y manejar herramientas informáticas convencionales.
- E8: Ser capaz de buscar y utilizar bibliografía en Física y otra bibliografía técnica, así como cualquier fuente de información relevante para trabajos de investigación y desarrollo técnico de proyectos.
- E9: Estar adecuadamente preparado para ejercitar una labor docente.
- E10: Ser capaz de mantenerse informado de los nuevos desarrollos.
- E13: Ser capaz de integrar los conocimientos recibidos de las diferentes áreas de la Física para la resolución de un problema.
- E15: Comprender y dominar el uso de los métodos matemáticos y numéricos más comúnmente utilizados.



3. Objetivos

Al finalizar la asignatura el alumno deberá ser capaz de:

- Manejar Linux a nivel de usuario
- Manejar un lenguaje de programación de nivel medio, en concreto el lenguaje C.
- Manejar programas de visualización de datos como los programas grace, xmakemol y opendx
- Ser capaz de elaborar programas en C de evaluación de funciones
- Ser capaz de elaborar programas en C para ajustar un conjunto de datos a una función y para interpolar un conjunto de datos
- Ser capaz de elaborar programas en C de solución de ecuaciones no lineales y diferenciales
- Ser capaz de elaborar programas en C de cálculo de integrales
- Comprender el método de Montecarlo y ser capaz de elaborar programas en C que usan el método de Montecarlo
- Comprender la dinámica molecular de un conjunto de átomos o moléculas a nivel básico y ser capaz de elaborar programas en C que simulan la dinámica clásica de un conjunto de átomos o moléculas.



4. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teóricas en aula	20	Estudio autónomo y resolución de problemas	20
Clases prácticas en aula de informática	65	Preparación y redacción de trabajos y ejercicios	25
Trabajos y ejercicios	15		
Exámenes	5		
Total presencial	105	Total no presencial	45



5. Bloques temáticos

a. Contextualización y justificación

La Física Computacional tiene diez temas. Los tres primeros temas son sobre Linux, programación en C y visualización de datos y funciones mediante programas. Los siguientes temas son sobre métodos numéricos y sus aplicaciones a problemas de Física. El uso de Linux y la programación en C y los programas de visualización de datos y funciones están presentes en todos los temas y por tanto, no parece conveniente hacer una agrupación de los diez temas en bloques temáticos. En su lugar hemos optado por presentar los diez temas formando parte de un único bloque.

b. Contenidos

Primer cuatrimestre (30 horas presenciales)

Tema 1. Introducción a Linux (6 horas presenciales)

- 1.1 Introducción: ¿Qué es un sistema operativo? ¿Qué es Linux? Distribuciones de Linux
- 1.2 Navegación por el sistema de archivos
- 1.3 Manipulación de archivos y directorios
- 1.4 Miscelánea
- 1.5 Visualización y edición de archivos
- 1.6 Redirecciones y tuberías
- 1.7 Control de procesos
- 1.8 Búsquedas: los comandos find y grep
- 1.9 Nociones básicas de awk

Tema 2. Programación básica en C (12 horas presenciales)

- 2.1 Introducción
- 2.2 Nociones básicas para creación de programas en C
- 2.3 Variables y constantes escalares
- 2.4 Operaciones o expresiones
- 2.5 Estructuras de decisión y/o control
- 2.6 Escritura y lectura de datos
- 2.7 Variables vectoriales o 'arrays'
- 2.8 Cadenas de caracteres o 'strings'
- 2.9 Funciones
- 2.10 Punteros
- 2.11 Operaciones de entrada y salida a ficheros
- 2.12 Compilado y enlazado de varios ficheros
- 2.13 Errores más frecuentes al programar en C
- 2.14 Estilo
- 2.15 Lenguajes basados en C



- 2.16 Bibliografía y referencias en Internet
- 2.17 Ejemplos prácticos de programación en C

Tema 3. Representación de datos y funciones mediante *gnuplot* (4 horas presenciales)

- 3.1 Introducción
- 3.2 Conceptos básicos
- 3.3 Modificación de los ejes
- 3.4 Modificación del aspecto de los datos
- 3.5 Títulos y leyendas
- 3.6 Exportar y guardar gráficos
- 3.7 Ajuste de datos por mínimos cuadrados mediante *gnuplot*
- 3.8 Ajuste de datos a una función arbitraria mediante *gnuplot*
- 3.9 Interpolación mediante splines cúbicos usando *gnuplot*

Tema 4. Ajuste e interpolación de datos (4 horas de clases presenciales)

- 4.1 Método de mínimos cuadrados para ajustar datos
- 4.2 Interpolación de Lagrange
- 4.3 Interpolación polinómica a trozos
- 4.4 Splines
- 4.5 Splines cúbicos

Tema 5. Ecuaciones no lineales (4 horas presenciales)

- 5.1 Método de bisección
- 5.2 Método de la secante
- 5.3 Método de la falsa posición
- 5.4 Método de Newton-Raphson

Aplicación de estos métodos a: El estado fundamental del deuterón, la ecuación de van der Waals,

Segundo cuatrimestre (30 horas presenciales)

Tema 6. Ecuaciones diferenciales (6 horas presenciales)

- 6.1 Método de Euler
- 6.2 Método del punto medio
- 6.3 Método de Runge-Kutta 2
- 6.4 Método de Runge-Kutta 4

Aplicación de estos métodos a: La desintegración radioactiva, el movimiento de una paracaidista, el movimiento de una barca en el agua, el átomo de hidrógeno y el equilibrio entre poblaciones de presas y depredadores, entre otros.

Tema 7. Integración numérica (4 horas presenciales)

- 7.1 Método del trapecio
- 7.2 Método de Simpson
- 7.3 Método de tres octavos de Simpson
- 7.4 Métodos compuestos: Trapecio y Simpson

Tema 8. Métodos de Monte Carlo (12 horas presenciales)

- 8.1 La aguja de Buffon



Guía docente de la asignatura

8.2 Origen de los métodos de Monte Carlo

8.3 Números pseudoaleatorios

8.4 Cálculo del número pi

8.5 Cálculo de integrales

8.6 La desintegración radioactiva

8.7 La difusión de partículas

8.8 La distribución de Boltzmann

8.9 El equilibrio térmico entre dos gases

8.10 El algoritmo de Metropolis

Tema 9. Dinámica Molecular (4 horas presenciales)

9.1 El potencial de Lennard-Jones

9.2 El algoritmo de Verlet

9.3 Aplicación a pequeños agregados de átomos

9.4 Introducción básica a xmakemol para visualizar agregados y dinámicas de agregados

Tema 10. Introducción a opendx (4 horas presenciales)

c. Métodos docentes

- Clases magistrales en el aula: Explicación de la teoría
- Clases prácticas en el aula de informática: El profesor explicará cómo se escribe, compila y ejecuta un programa en C y al mismo tiempo cada alumno escribirá, compilará y ejecutará el mismo programa.

d. Plan de trabajo

Las horas dedicadas a cada tema se especifican en el apartado b. Contenidos. Cada tema tiene el siguiente plan de trabajo y en este orden:

- ▲ Explicar en clase la teoría (34 % del tiempo)
- ▲ Resolver y explicar en el aula de informática la aplicación de la teoría a problemas (66 % del tiempo)

e. Evaluación

La evaluación de esta asignatura se basará en:

- Evaluación continua: Una vez al mes cada alumno resolverá un problema mediante un programa en C. Tendrá que escribir, compilar y ejecutar un programa en C. Enviará el programa en C y los ficheros de resultados, los cuales pueden ser figuras en jpg o ficheros de datos en ASCII, a través de la página Moodle de la asignatura.

Estos problemas mensuales sirven para evaluar de manera continua los conocimientos y capacidades de los alumnos y para que éstos vayan mejorando.

Los alumnos que entreguen todos los problemas podrán hacer el examen final a finales de mayo, antes del examen final oficial de junio. También podrán hacer el examen final oficial de junio para aprobar si no consiguieron aprobar en mayo o para subir nota.



Guía docente de la asignatura

- Examen final: Tendrá dos partes.

Parte teórica: Consistirá en responder por escrito a un test de respuesta múltiple. Durará unos 40 minutos y será el 40 % de la nota del examen final

Parte práctica: Consistirá en escribir, compilar y ejecutar dos programas en C. El alumno tendrá que entregar el programa en C y los ficheros de resultados antes del final de la parte práctica. Durará unas dos horas y será el 60 % de la nota del examen final.

Para aprobar la asignatura, la nota del examen final deberá ser igual o superior a 5 sobre 10.

f. Bibliografía básica

Enlaces en Internet:

- ⤴ <http://www.cprogramming.com/tutorial/c-tutorial.html>
- ⤴ <http://beginncprogram.blogspot.com.es>
- ⤴ https://es.wikibooks.org/wiki/Programación_en_C
- ⤴ [https://es.wikipedia.org/wiki/C_\(lenguaje_de_programación\)](https://es.wikipedia.org/wiki/C_(lenguaje_de_programación))
- ⤴ http://www.sisoft.ucm.es/Manuales/Lenguaje_C.pdf

- ⤴ Ránking TIOBE de lenguajes de programación. Se actualiza cada mes:
<http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>
- ⤴ Software y los lenguajes de programación usados para escribir ese software:
<http://www.lextrait.com/Vincent/implementations.html>

Libros sobre C:

- ⤴ Programación en C, Byron Gotfried, McGraw-Hill, Serie Schaum. México 1999

Hay once ejemplares de este libro en la planta baja de la biblioteca del campus Miguel Delibes

Signatura: 004 D3-Got.Pro

- ⤴ Fundamentos de programación en C, Miguel Ángel Rodríguez y Juan Manuel Sánchez Pérez, Manuales UEX, nº 41, Servicio de Publicaciones de la Universidad de Extremadura. Cáceres 2003

Hay un ejemplar de este libro en la planta baja de la biblioteca del campus Miguel Delibes

Signatura: 004 D3 C veg

- ⤴ Beginning C Programming, Ivor Horton, Wrox Press Ltd., Second Edition. Birmingham, UK 1997

Este libro no está en ninguna biblioteca de la UVa. Está en inglés y es uno de los libros que mejor explica C



g. Bibliografía complementaria

- ^ Lecture notes on “Computational Physics”. Morten Hjorth-Jensen, University of Oslo 2013.

Se trata de los apuntes del profesor Morten Hjorth-Jensen de la asignatura “Computational Physics”. Los apuntes están disponibles libremente en Internet.

- ^ A Survey of Computational Physics. Rubin H. Landau, Manuel José Páez and Cristian C. Bordeianu.

Princeton University Press, Princeton 2008. Signatura: C/Bc 53 LAN sur

Los contenidos de estos dos libros son de un nivel muy superior al de esta asignatura. Si algún alumno desea consultar estos libros, se recomienda que la consulta sea sobre un asunto muy concreto y no sobre un capítulo entero de estos libros.

h. Recursos necesarios

Un aula donde se impartirán las clases teóricas y varias aulas de informática con ordenadores conectados a Internet. Un sala de estudio donde los alumnos pueden resolver problemas individualmente y/o en grupo. Una biblioteca donde los alumnos pueden consultar libros, especialmente libros de problemas.

La plataforma **Moodle** es muy necesaria para diversas actividades de esta asignatura. En primer lugar, la página Moodle de la asignatura contiene el programa de la asignatura, la evaluación de la asignatura, los apuntes de cada tema en formato pdf y programas en C. Toda la información de esta Guía se encuentra en la página Moodle de la asignatura. En segundo lugar, los problemas mensuales y la parte práctica de los exámenes se envían a través de la página Moodle de la asignatura. Además, los alumnos hacen muchas consultas a través de la página Moodle.



6. Temporalización (por bloques temáticos)

BLOQUE TEMÁTICO	CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
1. Introducción al Linux	6 horas	Semanas 1 a 3
2. Programación básica en C	12 horas	Semanas 4 a 9
3. Representación de datos y funciones mediante grafice	4 horas	Semanas 10 a 11
4. Ajuste e interpolación de datos	4 horas	Semanas 12 a 13
5. Ecuaciones no lineales	4 horas	Semanas 14 a 15
6. Ecuaciones diferenciales	6 horas	Semanas 16 a 18
7. Integración numérica	4 horas	Semanas 19 a 20
8. Métodos de Monte Carlo	12 horas	Semanas 21 a 26
9. Dinámica molecular	4 horas	Semanas 27 a 28
10. Introducción a opendx	4 horas	Semanas 29 a 30

7. Sistema de calificaciones – Tabla resumen

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Evaluación continua: Problemas mensuales		Una vez al mes cada alumno tendrá que entregar un programa en C. Los alumnos que hayan entregado todos los problemas mensuales podrán hacer un examen final hacia finales de mayo, antes del examen final oficial de junio. También podrán hacer el examen final oficial de junio.
Examen final	100%	El examen final tendrá dos partes. Parte teórica: Consiste en responder por escrito a un test de respuesta múltiple. Parte práctica: Consiste en escribir y compilar dos programas en C sobre los temas estudiados en la asignatura. Nota del examen final: 40% nota parte teórica + 60% nota parte práctica Para aprobar hay que obtener una nota en el examen final igual o superior a 5 sobre 10
Examen extraordinario	100%	Las mismas normas de evaluación que tiene el examen final.