

Guía docente de la asignatura

Fecha última actualización: 21/06/2021

Fecha de aprobación: 21/06/2021

Informática Aplicada a la Bioquímica

Grado	Grado en Bioquímica	Rama	Ciencias
Módulo	Física, Matemática e Informática para las Biociencias Moleculares	Materia	Informática Aplicada a la Bioquímica
Curso	2º	Semestre	1º
Créditos	6	Tipo	Obligatoria

BREVE DESCRIPCIÓN DE CONTENIDOS (Según memoria de verificación del Grado)

- Conceptos básicos. Sistemas numéricos y lógica booleana. Datos. Tipos de datos. Cálculo computacional. Errores. Sistemas operativos.
- Concepto de programa. Diagrama de flujo. Lenguajes de programación. Algoritmos y estructuras de datos.
- Elementos de programación en C/Python (o entorno equivalente).
- Aplicación de la informática a la Bioquímica, la Biología Molecular.
- Algoritmos de diferenciación e integración numérica. Simulación virtual de procesos bioquímicos.
- Métodos de inteligencia artificial: redes neuronales y algoritmos genéticos. Aplicaciones.

COMPETENCIAS ASOCIADAS A MATERIA/ASIGNATURA

COMPETENCIAS GENERALES

- CG05 - Haber desarrollado las habilidades de aprendizaje necesarias para emprender estudios posteriores de especialización con un alto grado de autonomía, incluyendo la capacidad de asimilación de las distintas innovaciones científicas y tecnológicas que se vayan produciendo en el ámbito de las Biociencias Moleculares

COMPETENCIAS ESPECÍFICAS

- CE24 - Poseer las habilidades matemáticas, estadísticas e informáticas para obtener, analizar e interpretar datos, y para entender modelos sencillos de los sistemas y procesos biológicos a nivel celular y molecular
- CE25 - Saber buscar, obtener e interpretar la información de las principales bases de datos biológicos (genómicos, transcriptómicos, proteómicos, metabolómicos y similares derivados de otros análisis masivos) y de datos bibliográficos, y usar las herramientas bioinformáticas básicas



COMPETENCIAS TRANSVERSALES

- CT01 - Adquirir la capacidad de razonamiento crítico y autocrítico
- CT04 - Tener capacidad de aprendizaje y trabajo autónomo
- CT06 - Saber reconocer y analizar un problema, identificando sus componentes esenciales, y planear una estrategia científica para resolverlo
- CT07 - Saber utilizar las herramientas informáticas básicas para la comunicación, la búsqueda de información, y el tratamiento de datos en su actividad profesional

RESULTADOS DE APRENDIZAJE (Objetivos)

SABER:

- Entender las bases de las ciencias de la computación e informática.

SABER HACER:

- Expresarse correctamente con términos informáticos básicos.
- Saber manejarse bien en el entorno de diferentes sistemas operativos para las operaciones básicas (Windows, MacOS y Linux)
- Saber realizar con soltura y bien las operaciones básicas con un ordenador: elementos básicos de la pantalla de inicio, manejo de ventanas, documentos y carpetas, directorios, formateado de discos, copiado de documentos. Conexión de periféricos. Destreza en el uso del ratón. Herramientas básicas de mantenimiento del disco duro y mantenimiento del ordenador en buen funcionamiento. Copias de seguridad (Back-ups).
- Utilizar con soltura y bien las herramientas básicas de Internet. Buscadores y organización de búsquedas. Correo electrónico.
- Saber utilizar con soltura y bien un procesador de texto, un editor de imágenes y un programa de presentación de charlas y resultados. Saber manejar los diferentes formatos de los documentos y su transformación en documentos de reconocimiento por software libre o universal (rtf, tiff, pdf, etc).
- Saber realizar bien búsquedas en las principales bases de datos bibliográficos (Medline y similares).
- Saber interpretar correctamente los resultados que se generan de las bases de datos biológicos tras una búsqueda de alineamiento de proteínas y ácidos nucleicos (Blast, Fasta o similares)
- Desarrollar programas sencillos de aplicación en MATLAB, (Visual) BASIC, (Visual) C/Python o similares

PROGRAMA DE CONTENIDOS TEÓRICOS Y PRÁCTICOS

TEÓRICO

- Tema 1. Introducción a la Informática
- Tema 2. Fundamentos de programación
- Tema 3. Introducción a la Programación en Python 3.x
- Tema 4. Estructuras de Control
- Tema 5. Tipos de Datos Estructurados
- Tema 6. Programación Modular
- Tema 7. Introducción a las Bases de Datos



- Tema 8. Inteligencia Artificial y Bioquímica
- Tema 9. Algoritmos Bioinspirados

PRÁCTICO

Seminarios/Talleres (0,2 ECTS/ 5hs)

- Seminario 1. Introducción a los Sistemas Operativos (Windows, Unix/Linux)
- Seminario 2. Introducción al entorno de programación de Python
- Seminario 3. Herramientas para la Diferenciación e Integración Numérica
- Seminario 4. Bases de Datos Biológicas
- Seminario 5. Introducción al módulo Biopython para Bioinformática

Prácticas de Laboratorio (1,2 ECTS/ 30 h):

- Práctica 1. Sistemas Operativos
- Práctica 2. Primeros Programas en Python
- Práctica 3. Estructuras de Control
- Práctica 4. Tipos de Datos Estructurados (uso del módulo Biopython) – Ejemplo de uso con estructuras del PDB, Protein Data Bank
- Práctica 5. Creación de Módulos – Ejemplo de uso con estructuras del PDB
- Práctica 6. Algoritmos de Diferenciación e Integración Numérica

BIBLIOGRAFÍA

BIBLIOGRAFÍA FUNDAMENTAL

- J.G. Molina, F.M. Dato, et al., Una introducción a la Programación. Un enfoque algorítmico, Thompson, 2005.
- Traducción al español de la última versión del libro de Mark Pilgrim sobre **Python 3**, Dive Into Python 3 - Inmersión en Python 3, por José Miguel González Aguilera: <https://storage.googleapis.com/google-code-archive-downloads/v2/code.google.com/inmersionenpython3/inmersionEnPython3.0.11.pdf>
- How to Think Like a Computer Scientist: Interactive Edition, Runestone Interactive Project at Luther College, led by Brad Miller and David Ranum. Based on the original work by J. Elkner, A.B. Downey and C. Meyers, How to Think Like a Computer Scientist: Learning with Python, 2nd Edition (Python 2.x), 2012: <http://interactivepython.org/runestone/static/thinkcspy/index.html>
- Al Sweigart, Automate the Boring Stuff with Python. Practical programming for total beginners. Free under a Creative Commons license: <https://automatetheboringstuff.com/#toc> (Python 3.3)
- Lahoz-Beltra, Rafael, Bioinformática: simulación, vida artificial e Inteligencia Artificial, López de Santos, 2004.

BIBLIOGRAFÍA COMPLEMENTARIA

- Video “Curso Python desde 0” (**Python 3.6**) <https://www.youtube.com/playlist?list=PLU8oAlHdN5BlvPxziopYZRd55pdqFwkeS>
- E. Rich, K. Knight, Inteligencia Artificial, segunda edición, Mc Graw Hill Co. 1992.
- S. Russell, P. Norvig, Inteligencia Artificial: un Enfoque Moderno, 2ª Edición, Ed. Pearson-



Prentice Hall, 2004.

- Pons O., Marín N., Medina J.M., Acid S., Vila M.A., Introducción a las Bases de Datos: El modelo Relacional, Thomson Paraninfo, 2005.
- James A. Goodrich; Jennifer F. Kugel, Binding and Kinetics for Molecular Biologists, Spiral bound, Cold Spring Harbor Laboratory Press
- Uri Alon , An Introduction to Systems Biology : Design Principles of Biological Circuits, CRC Press, 2006

ENLACES RECOMENDADOS

- <http://biopython.org/wiki/Biopython>
- <http://www.practicepython.org/> (Beginner Python exercises)
- http://www.open-bio.org/wiki/Main_Page
- http://bioperl.org/wiki/Main_Page

METODOLOGÍA DOCENTE

- MD01 Lección magistral/expositiva
- MD03 Prácticas de laboratorio y/o informática
- MD07 Actividad no presencial de aprendizaje mediante el estudio de la materia, el análisis de documentos, la elaboración de memorias...

EVALUACIÓN (instrumentos de evaluación, criterios de evaluación y porcentaje sobre la calificación final)

EVALUACIÓN ORDINARIA

Se llevará a cabo una evaluación mixta que incluye la realización de un test de conocimientos adquiridos, la entrega y defensa de una colección de ejercicios resueltos durante las clases prácticas y un examen de programación básica, consistente en el desarrollo controlado de uno o varios de los ejercicios resueltos durante el curso por el estudiante.

- Teoría - Prueba tipo test de conocimientos: 40% (competencias evaluadas CG5,CB5,CE24,CE25,CT6)
- Teoría - Prueba de programación: 30 % (competencias evaluadas CG5,CB5,CE24,CT1,CT6,CT7)
- Prácticas - Colección de ejercicios resueltos: 30 % (competencias evaluadas CG5,CB5,CE24,CE25,CT1,CT5,CT6,CT7)

EVALUACIÓN EXTRAORDINARIA

En el examen extraordinario, el 30% de la "Colección de ejercicios resueltos" (prácticas realizadas durante el curso) quedará incluida dentro del "Examen de programación", que en tal caso será un 60% de la nota y que por tanto incluirá ejercicios del mismo tipo que los resueltos en la "Colección de ejercicios resueltos". De esta manera la estructura del examen extraordinario es la siguiente: "Prueba tipo test de conocimientos" (40%) y "Examen de programación" (60%).



EVALUACIÓN ÚNICA FINAL

- Según la Normativa de Evaluación y de Calificación de los Estudiantes de la Universidad de Granada (modificada en Consejo de Gobierno el 26 de octubre de 2016) “la evaluación será preferentemente continua, entendiéndose por tal la evaluación diversificada que se establezca en las Guías Docentes de las asignaturas. No obstante, las Guías Docentes contemplarán la realización de una evaluación única final a la que podrán acogerse aquellos estudiantes que no puedan cumplir con el método de evaluación continua por motivos laborales, estado de salud, discapacidad, programas de movilidad o cualquier otra causa debidamente justificada que les impida seguir el régimen de evaluación continua” (artículo 6.2).

“Para acogerse a la evaluación única final, el estudiante, en las dos primeras semanas de impartición de la asignatura, o en las dos semanas siguientes a su matriculación si ésta se ha producido con posterioridad al inicio de la asignatura, lo solicitará, a través del procedimiento electrónico, al Director del Departamento o al Coordinador del Máster, alegando y acreditando las razones que le asisten para no poder seguir el sistema de evaluación continua. En el caso de asignaturas de grado con docencia compartida por varios Departamentos, el estudiante lo solicitará a cualquiera de los Departamentos implicados. El Director del Departamento o el Coordinador del Máster al que se dirigió la solicitud, oído el profesorado responsable de la asignatura, resolverá la solicitud en el plazo de diez días hábiles. Transcurrido dicho plazo sin que el estudiante haya recibido respuesta expresa por escrito, se entenderá estimada la solicitud. En caso de denegación, el estudiante podrá interponer, en el plazo de un mes, recurso de alzada ante el Rector, quien podrá delegar en el Decano o Director del Centro o en el Director de la Escuela Internacional de Posgrado, según corresponda, agotando la vía administrativa. No obstante lo anterior, por causas excepcionales sobrevenidas y justificadas (motivos laborales, estado de salud, discapacidad, programas de movilidad, representación o cualquier otra circunstancia análoga), podrá solicitarse la evaluación única final fuera de los citados plazos, bajo el mismo procedimiento administrativo”. (NCG127/2: Instrucción relativa a la aplicación del artículo 8.2). La normativa se puede consultar en <https://www.ugr.es/sites/default/files/2017-09/examenes.pdf>.

- La evaluación única final se realizará en un solo acto académico el mismo día y hora de la convocatoria oficial de examen ordinario para la asignatura (es decir, se realiza de forma conjunta). Dicha prueba (evaluada de 0 a 10) tendrá la misma estructura de un examen extraordinario (40-60%) teniendo en cuenta que las preguntas del "Examen de programación" (60%) deben suplir el trabajo realizado durante el curso mediante la "Colección de ejercicios resueltos" y garantizando que el alumno ha adquirido la totalidad de las competencias descritas en esta misma guía docente.

INFORMACIÓN ADICIONAL

Siguiendo las recomendaciones de la CRUE y del Secretariado de Inclusión y Diversidad de la UGR, los sistemas de adquisición y de evaluación de competencias recogidos en esta guía docente se aplicarán conforme al principio de diseño para todas las personas, facilitando el aprendizaje y la demostración de conocimientos de acuerdo a las necesidades y la diversidad funcional del alumnado.

