



DESCRIPCIÓN DE LA ASIGNATURA

Grado/Máster en:	Graduado/a en Ingeniería de la Salud por la Universidad de Málaga
Centro:	Escuela Técnica Superior de Ingeniería Informática
Asignatura:	Fundamentos de la Programación
Código:	103
Tipo:	Formación básica
Materia:	INFORMÁTICA
Módulo:	MÓDULO DE FORMACIÓN BÁSICA
Experimentalidad:	69 % teórica y 31 % práctica
Idioma en el que se imparte:	Español
Curso:	1
Semestre:	1
Nº Créditos	6
Nº Horas de dedicación del estudiante:	150
Nº Horas presenciales:	60
Tamaño del Grupo Grande:	72
Tamaño del Grupo Reducido:	30
Página web de la asignatura:	

EQUIPO DOCENTE

Departamento: LENGUAJES Y CIENCIAS DE LA COMPUTACIÓN

Área: LENGUAJES Y SISTEMAS INFORMÁTICOS

Nombre y Apellidos	Mail	Teléfono Laboral	Despacho	Horario Tutorías
Coordinador/a: JUAN ANTONIO FALGUERAS CANO	jfalgueras@uma.es	952132792	3.2.32 - E.T.S. Ing. Telecomunicación	Todo el curso: Lunes 09:00 - 11:00, Miércoles 12:45 - 14:45, Martes 10:00 - 12:00
ISMAEL NAVAS DELGADO	ismael@uma.es	952136314	3.2.49 - E.T.S.I. INFORMÁTICA	Primer cuatrimestre: Martes 08:30 - 10:30, Jueves 12:30 - 14:30, Jueves 10:30 - 12:30
MARIA VICTORIA BELMONTE MARTINEZ	mvelmonte@uma.es	952132809	3.2.36 - E.T.S.I. INFORMÁTICA	Primer cuatrimestre: Lunes 09:30 - 12:45, Miércoles 12:30 - 15:15

RECOMENDACIONES Y ORIENTACIONES

Aunque no se pide ningún nivel previo de conocimientos sobre los contenidos de esta asignatura, para su seguimiento es conveniente un mínimo grado de destreza en el uso de sistemas informáticos: experiencia en el acceso a internet, uso de correo electrónico o edición de textos.

Se darán conocimientos básicos al comienzo del curso de acceso a Terminal UNIX/Linux y se irán usando a lo largo del curso. Esta introducción al Terminal UNIX/Linux está justificada por la importancia de su uso actual tanto en investigación en bioinformática como en gestión de sistemas, sentando las bases del funcionamiento real, no gráfico, de los sistemas operativos.

Siendo la resolución algorítmica de problemas en un lenguaje informático sobre el ordenador novedosa como método de trabajo y el principal objetivo del curso, la práctica mediante ejercicios será mucho más eficiente si el alumno o bien tiene destreza informática previa o inmediatamente, al principio del curso, el alumno instala y aprende a usar estas herramientas de desarrollo.

CONTEXTO

Hoy día es fundamental en cualquier actividad profesional competitiva no sólo el uso cotidiano de la tecnología informática sino el diseño de nuevos usos para su incorporación a las actividades de comunicación, gestión e investigación. Como ingenieros de profesiones relacionadas con la salud, el uso y comprensión del funcionamiento de los sistemas informáticos toman una relevancia aún mayor. No es suficiente el uso de aplicaciones con interfaces gráficas de usuario sencillas, adecuadas a tareas específicas, sino que, como parte de la actividad de un ingeniero, es imprescindible construir nuevos programas para el análisis y gestión de datos.

La programación de ordenadores amplía y profundiza nuestro conocimiento sobre la mecánica y funcionamiento interno de los sistemas informáticos, sus sistemas operativos y posibles tipos de aplicaciones. El programador adquiere los fundamentos para hacerse consciente de las posibilidades y costo computacional de los proyectos antes de iniciarlos. Desarrolla además, paralelamente, un tipo de destreza cognitiva apropiada a un mejor uso de los sistemas informáticos en general, haciéndonos mejores y más capaces usuarios de las nuevas tecnologías.

Esta asignatura utiliza lenguajes de programación actuales usados en el área de la Ingeniería de la Salud sin depender de las particulares características de ninguno de ellos. No obstante, dentro del inevitable compromiso con un lenguaje concreto de programación, se usará un lenguaje de programación de entre los de más auge y uso en el contexto de la Ingeniería de la Salud.

El temario, sin embargo es altamente independiente del lenguaje de programación, introduciendo conceptos fundamentales comunes a todos los lenguajes de programación. Se trata de un primer acercamiento del alumno al mundo de la programación y, por ello, se aportan los elementos de programación básicos, incluyendo la forma rigurosa y estructurada de razonar algorítmicamente. En particular, se hace especial hincapié en las técnicas correctas de escritura de programas y se inicia al alumno en técnicas de programación estructurada, diseño descendente, abstracción procedimental y uso de datos y estructuras de datos fundamentales.

COMPETENCIAS



1 Competencias generales y básicas.

- 1.3** CB3 Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.
- 1.4** CB4 Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.
- 1.11** CG06 Capacidad para identificar, formular y resolver problemas dentro de contextos amplios y multidisciplinares en los campos de la ingeniería y las ciencias de la salud, mediante la integración de conocimientos y la participación en equipos multidisciplinares.
- 1.13** CG08 Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.
- 1.15** CG10 Capacidad para comunicar y transmitir los conocimientos y conclusiones en el ámbito de la ingeniería de la salud, a público especializado y no especializado, de un modo claro y preciso.

2 Competencias específicas. Específicas de Titulación

- 2.3** CEB03 Capacidad para comprender y dominar los conceptos básicos de la algorítmica y los lenguajes de programación, y su aplicación para la resolución de problemas propios de la ingeniería.
- 2.4** CEB04 Capacidad para comprender y aplicar los conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

CONTENIDOS DE LA ASIGNATURA

Un bloque único de temas

- 1. Conceptos básicos de Informática
 - 1.1. Presentación de la asignatura. Conceptos de informática
 - 1.2. Sistema operativo, compiladores. Instalación del compilador
 - 1.3. Algoritmo. Concepto de algoritmo, primitiva, procesador y entorno
 - 1.4. Léxico, sintaxis, semántica
- 2. Variables
 - 2.1. Formato de los programas
 - 2.2. Tipos de datos simples, variables, constantes
 - 2.3. Expresiones, prioridad de operadores y conversión de tipos
 - 2.4. Codificación de datos, entrada/salida
- 3. Estructuras de control
 - 3.1. Variables booleanas
 - 3.2. Ejemplos de estructuras de selección
 - 3.3. Estructuras iterativas: while, for
- 4. Subprogramas
 - 4.1. Subprogramas. Subprogramación
 - 4.2. Declaración de subprogramas e interfaces correctas
 - 4.3. Variables locales, ámbitos de visibilidad, paso de parámetros
 - 4.4. Ejemplos y ejercicios de subprogramación
- 5. Listas
 - 5.1. Listas, definición, creación y acceso
 - 5.2. Listas y subprogramas
 - 5.3. Esquemas existe, paraTodo, contador, suma, máximo y mínimo
 - 5.4. Búsqueda y filtro
- 6. Cadenas de caracteres
 - 6.1. Funciones específicas
 - 6.2. Codificaciones
 - 6.3. Búsqueda en cadenas de caracteres
- 7. Diccionarios



- 7.1. Creación y uso de paquetes de datos
- 7.2. Uso con subprogramas
- 7.3. Ejemplos
- 8. Ficheros
 - 8.1. Tipos y gestión de ficheros
 - 8.2. Uso de ficheros de texto
 - 8.3. Lectura de ficheros en estructuras

ACTIVIDADES FORMATIVAS

Actividades presenciales

Actividades expositivas

Lección magistral

Actividades prácticas en instalaciones específicas

Prácticas en laboratorio

ACTIVIDADES DE EVALUACIÓN

Actividades de evaluación presenciales

Actividades de evaluación del estudiante

Examen parcial

Examen final

RESULTADOS DE APRENDIZAJE / CRITERIOS DE EVALUACIÓN

Como resultados de aprendizaje se esperan las siguientes capacidades:

- RA1. Capacidad para comprender y aplicar los conceptos básicos de la algorítmica y los lenguajes de programación, y su aplicación para la resolución de problemas propios de la ingeniería
- RA2. Capacidad para comprensión de las etapas y complejidad de los pasos en la ejecución de los programas, la evaluación cualitativa de su eficiencia temporal y los requisitos computacionales de su ejecución
- RA3. Capacidad para diseñar programas para la resolución de problemas y construcción de las estructuras de datos iniciales adecuadas para cada problema
- RA4. Capacidad para la resolución mediante programas de gestión administrativa informática
- RA5. Capacidad para el desarrollo de programas para el análisis de datos manipulando ficheros de datos
- RA6. Capacidad para la resolución de los problemas mediante métodos numéricos; algorítmica numérica; estadística y optimización
- RA7. Capacidad para comprender y aplicar los conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería

Los anteriores resultados de aprendizaje cubren las Competencias Específicas fijadas de la siguiente forma:

RA1, RA2, RA3, RA4, RA5 y RA6 cubren CG08, CEB01, CEB03 y CEB04. RA7 cubre CG06 y CG10

Como criterios de evaluación:

- CE1. La correcta escritura de programas tanto a nivel de claridad y formato, como de sencillez de código
- CE2. Uso adecuado de las estructuras de control, bucles y decisiones.
- CE3. Elección conveniente de las estructuras de datos
- CE2. La correcta utilización de las herramientas de programación y sistema de desarrollo
- CE3. La capacidad de resolución de nuevos problemas
- CE4. La evidencia de comprensión de los mecanismos usados y de posibles alternativas
- CE5. La capacidad para el modelado algorítmico de situaciones reales en gestión, cálculo numérico y análisis de datos bioinformáticos

Con estos criterios de evaluación se cubren: RA1, RA4, RA5 y RA6 con CE5, RA2 y RA3 con CE4, RA3 con CE2 y CE3 y, finalmente, RA7 con CE1, CE2 y CE3

PROCEDIMIENTO DE EVALUACIÓN

La evaluación de las competencias asociadas a la asignatura se hará a partir del seguimiento de la actividad del alumno a lo largo del curso, de su evolución y de los resultados tanto de las prácticas presenciales, no presenciales y controles o exámenes parciales y no presenciales durante el periodo docente y, al final del curso, de un examen global sobre todo el contenido desarrollado.



Para el seguimiento de la actividad durante el periodo docente se realizarán actividades de tutorización personalizada, seminarios específicos, especialmente al principio del curso para ayudar a la instalación y uso de las nuevas tecnologías a usar y, a lo largo de todo el curso, mediante el seguimiento de su participación en los laboratorios prácticos de la asignatura y los foros y actividades on-line del Campus Virtual de la asignatura.

Se realizarán controles o exámenes parciales en los momentos previos a los cambios de contenido y después de suficientes actividades prácticas y de la realización de ejercicios tanto prácticos como teóricos relacionados sobre los contenidos a examinar.

El examen final, tras el periodo docente, abordará todos los contenidos y destrezas desarrollados durante todo el curso y tendrá un peso considerablemente mayor sobre la nota final. Este examen final permitirá además recuperar cualquier calificación previa insuficiente.

El peso en la nota final de cada parte será el siguiente:

- La evaluación continua tendrá hasta un máximo de 3 puntos.
- El examen final completará hasta 10 puntos, o sea, tendrá un valor de 10 menos la calificación en la evaluación continua.

Así por ejemplo, un alumno que tenga una evaluación continua calificada con 0 puntos, tendrá opción a ser evaluado sobre 10 en el examen final; un alumno que alcance con calificación continua de 3 puntos, podrá acudir al examen final para cubrir el resto de hasta 7 puntos más para alcanzar la nota final de 10.

Estos criterios de evaluación se mantendrán vigente durante la evaluación ordinaria (febrero y septiembre), pero no ya en las sucesivas evaluaciones extraordinarias, donde la calificación del examen único a realizar constituirá el 100% de la nota final.

Los alumnos a tiempo parcial y los alumnos deportistas de alto nivel universitarios se considerarán en principio alumnos que no siguen el proceso de evaluación continua. No obstante, en función de su situación particular, pueden optar por el proceso de evaluación continua.

En cualquier caso, tras la realización de cualesquiera de las pruebas, el profesor se reserva el derecho a convocar al alumno para comprobar la autoría de las respuestas o prácticas entregadas, reservándose así el derecho de modificar la calificación obtenida por un alumno, tanto en los controles como en los exámenes finales, mediante la correspondiente revisión en presencia del mismo.

El plagio, total o parcial, de las prácticas no presenciales, en alguna de sus pruebas de autoría asociadas o del examen final ordinario, conllevará una valoración de 0 puntos en el total final de la convocatoria ordinaria.

La calificación de Matrícula de Honor correspondería a una evaluación máxima o justificadamente cercana a la máxima en todas las evaluaciones parciales acompañado de un seguimiento constante y previsor de los contenidos de la asignatura, así como de la realización de todas las prácticas y trabajos extraordinarios y de una participación activa durante todo el curso y la cooperación con los compañeros de clase. El número máximo de Matrículas de Honor viene estipulado por criterios propios de la Escuela.

BIBLIOGRAFÍA Y OTROS RECURSOS

Básica

Introduction to Computation and Programming using Python (second edition); J.V. Guttag; Edit. The MIT Press; 2015.

Introduction to Computer Science Using Python: A Computational Problem-Solving Focus.; C. Dierbach; Edit. Wiley; 2012.

Learning Python; M. Lutz; Edit. O'Reilly; 2013

Complementaria

Introducción a la programación con Python (<http://www.mclibre.org/consultar/python/>)

Learn Python (oficial) (en español) (<http://www.learnpython.org/es/>)

New Boston: (disponible online: <https://thenewboston.com/videos.php?cat=98>)

Python for Everybody. Exploring Data Using Python 3; C.R. Severance; 2016 (disponible online: http://do1.dr-chuck.com/pythonlearn/EN_us/pythonlearn.pdf)

Python para principiantes (<http://librosweb.es/libro/python/>)

Tutorial de Python (<http://docs.python.org.ar/tutorial/3/>)

Tutorials Point: (disponible online: <https://www.tutorialspoint.com/python3/>)

DISTRIBUCIÓN DEL TRABAJO DEL ESTUDIANTE

ACTIVIDAD FORMATIVA PRESENCIAL

Descripción	Horas	Grupo grande	Grupos reducidos
Lección magistral	41,4	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Prácticas en laboratorio	18,6	<input type="checkbox"/>	<input checked="" type="checkbox"/>

TOTAL HORAS ACTIVIDAD FORMATIVA PRESENCIAL 60

TOTAL HORAS ACTIVIDAD FORMATIVA NO PRESENCIAL 75

TOTAL HORAS ACTIVIDAD EVALUACIÓN 15



TOTAL HORAS DE TRABAJO DEL ESTUDIANTE ~~150~~ horas Grupo grande Grupos reducidos

