



## Teaching Guide

Identifying Data					2018/19
Subject (*)	Introduction to programming	Code	614522001		
Study programme	Mestrado Universitario en Bioinformática para Ciencias da Saúde				
Descriptors					
Cycle	Period	Year	Type	Credits	
Official Master's Degree	1st four-month period	First	Optional	6	
Language	Spanish				
Teaching method	Face-to-face				
Prerequisites					
Department	Computación				
Coordinador	Cabrero Canosa, Mariano Javier	E-mail	mariano.cabrero@udc.es		
Lecturers	Cabrero Canosa, Mariano Javier	E-mail	mariano.cabrero@udc.es		
Web	moodle.udc.es				
General description	In this subject it is intended that students without training in programming acquire the basic notions for carrying out programs. The Python programming language will be used: different types of data and the basic control structures that are used to make a application.				

## Study programme competences / results

Code	Study programme competences / results
A3	CE3 ? To analyze, design, develop, implement, verify and document efficient software solutions based on an adequate knowledge of the theories, models and techniques in the field of Bioinformatics
B1	CB6 - Own and understand knowledge that can provide a base or opportunity to be original in the development and/or application of ideas, often in a context of research
B5	CB10 - Students should possess learning skills that allow them to continue studying in a way that will largely be self-directed or autonomous.
B8	CG3 - Be able to work in a team, especially of interdisciplinary nature
C3	CT3 - Use the basic tools of the information technology and communications (ICT) necessary for the exercise of their profession and lifelong learning
C6	CT6 - To assess critically the knowledge, technology and information available to solve the problems they face to.

## Learning outcomes

Learning outcomes	Study programme competences / results		
Interiorizar as boas prácticas de programación.	AJ3	BJ5 BJ8	
Usar as estruturas de datos adecuadas e programar os algoritmos de manipulación para solucionar problemas reais.	AJ3	BJ1 BJ8	
Capacidade para realizar programas sinxelos no computador empregando unha linguaxe de alto nivel.	AJ3	BJ1 BJ5 BJ8	CJ3 CJ6
Ser capaz de deseñar, avaliar, comparar e analizar solucións algorítmicas básicas a problemas usuais en Bioinformática.	AJ3	BJ1	CJ6

## Contents

Topic	Sub-topic



1. Introducción	<ul style="list-style-type: none"> <li>a. Algoritmos. Representación. Accións primitivas/no primitivas</li> <li>b. Programas. Proceso de construción</li> <li>c. Linguaxes de programación: máquina, baixo nivel, alto nivel</li> <li>d. Compiladores. Intérpretes</li> <li>e. Entornos de desenvolvemento e ferramentas: Python</li> </ul>
2. Conceptos básicos	<ul style="list-style-type: none"> <li>a. Estructura de un programa</li> <li>b. Constantes, Variables.</li> <li>c. Tipos de datos: enteiro, real, lóxico, carácter, ?</li> <li>d. Estructuras simples: listas (arrays), cadenas, ?</li> <li>e. Operadores y expresiones (aritméticas, lógicas)</li> <li>f. Declaración de variables e constantes</li> <li>g. Entrada y salida estándar</li> </ul>
3. Sentencias de control	<ul style="list-style-type: none"> <li>a. Secuencial</li> <li>b. Alternativa</li> <li>c. Repetitiva: while, for</li> </ul>
4. Funcións	<ul style="list-style-type: none"> <li>a. Definición, declaración e chamada de función</li> <li>b. O ámbito das variables</li> <li>c. Paso de argumentos</li> <li>d. Recursividad</li> <li>e. Módulos</li> </ul>
5. Ficheiros	<ul style="list-style-type: none"> <li>a. Apertura e peche</li> <li>b. Lectura e escritura de datos</li> <li>c. Acceso directo aos datos</li> </ul>
6. Introducción a estruturas abstractas	<ul style="list-style-type: none"> <li>a. Listas</li> <li>b. Pilas</li> <li>c. Colas</li> <li>d. Árbores</li> </ul>
6. Introducción á orientación a obxectos	<ul style="list-style-type: none"> <li>a. Clases</li> <li>b. Obxectos</li> <li>c. Propiedades</li> <li>d. Métodos</li> <li>e. Concepto de herencia</li> </ul>
7. Excepcións	<ul style="list-style-type: none"> <li>a. Tipos</li> <li>b. Captura</li> <li>c. Lanzamento</li> <li>d. Creación</li> </ul>
8. Librerías científicas en Python	<ul style="list-style-type: none"> <li>a. SciPy</li> <li>b. NumPy</li> <li>c. Matplotlib</li> <li>d. BioPython</li> </ul>

Planning				
Methodologies / tests	Competencies / Results	Teaching hours (in-person & virtual)	Student's personal work hours	Total hours
Problem solving	A3 B8 C3 C6	19	57	76
Mixed objective/subjective test	A3	2	6	8
Case study	A3 B1 B5	0	19	19
Guest lecture / keynote speech	A3 B1 B5	11	33	44
Personalized attention		3	0	3

(\*)The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.



## Methodologies

Methodologies	Description
Problem solving	Esta actividade suporá o estudo de casos prácticos e exemplos ademáis da realización de distintos exercicios de programación. Co fin de afianzar os conceptos teóricos presentaranse supostos prácticos, que nun principio serán resoltos polo profesor para que orienten os alumnos. A medida que se avance no desenvolvemento teórico formularase a resolución de problemas por parte dos alumnos. A proposta de actividades estará dispoñible ao alumno con suficiente antelación. O labor do profesor será a supervisión solucionando dúbidas e corrigindo erros de interpretación, malos hábitos de programación, erros de sintaxe, etc.
Mixed objective/subjective test	Avaliación sumativa do alumno mediante un exame escrito cunha parte teórica con preguntas tipo test e unha parte práctica para resolver pequenos problemas de programación. A proba tratará de medir se o alumno adquiriu os conceptos fundamentais de programación e adestrouse o suficiente como para posuír as habilidades precisas para resolver supostos prácticos. O alumno poderá facer uso do ordenador para, ademais de contestar ás preguntas, consultar dúbidas acerca da sintaxe concreta de algún comando.
Case study	Actividade non presencial para afondar nos conceptos fundamentais da materia. Consistirá no estudo persoal do alumno, a través do material suxerido e proporcionado polo profesor. A proporción de uso desta metodoloxía será maior fronte a sesión maxistral cando o número de estudantes sexa baixo e será acordado con estes.
Guest lecture / keynote speech	Actividade presencial para expoñer conceptos fundamentais da materia. Consistirá na exposición oral do profesor apoiada con medios multimedia. Durante a presentación tratarase de interactuar co alumno formulando preguntas dirixidas co fin de afianzar conceptos e facilitar a aprendizaxe. A proporción de uso desta metodoloxía será maior fronte a estudo de casos cando o número de estudantes sexa alto e será acordado con estes.

## Personalized attention

Methodologies	Description
Problem solving	É fundamental a atención ao alumno para resolver cantas dúbidas de concepto ou de procedemento poidan xurdir durante a resolución dos supostos prácticos. Prestarase especial atención a aqueles alumnos que presenten maiores dificultades na súa aprendizaxe co fin de que o seu progreso non se vexa retardado respecto ao xeral do resto de estudantes.

## Assessment

Methodologies	Competencies / Results	Description	Qualification
Problem solving	A3 B8 C3 C6	Valorarase a participación do alumno así como a realización de diversos traballos puntuables que se detallarán durante o curso e que poderán resolverse na clase ou na titoría. Non é necesario entregar todos os traballos para aprobar, aínda que sí para conseguir a máxima nota.	65
Mixed objective/subjective test	A3	Realización obrigatoria. Necesario aprobar o exame para superar a materia. O exame constará dunha parte tipo test (40% da nota final) e unha parte práctica (60%).	35

## Assessment comments



## Traballos prácticos

- Soamente os alumnos con cualificación de NON APTO ou NON PRESENTADO en prácticas na primeira oportunidade poderán entregar os traballos propostos para a segunda oportunidade.

- De acordo ao artigo 14, apartado 4, da normativa\*, o plaxio dos traballos prácticos levará unha nota global de NON APTO, tanto ao estudante que presente material copiado como ao que o facilitara, e por tanto a cualificación de SUSPENSO na convocatoria anual.

## Primeira e segunda oportunidade

- As cualificacións obtidas en actividades de solución de problemas serán válidas tan só para o curso académico no que se realicen. Manteranse para aqueles alumnos suspensos ou non presentados na primeira oportunidade. Só aqueles con prácticas SUSPENSAS terán a opción de realizar de novo as prácticas.

## Matrícula a tempo parcial

- Os alumnos matriculados a tempo parcial terán que entregar as actividades avaliadas nas condicións e prazos específicos que se establecerán. Será obrigación do estudante comunicar a súa situación ao profesorado.

## Non presentado

- Terá a condición de Non presentado (NP) quen non presente ningún traballo práctico nin concorra á proba obxectiva no período oficial de avaliación.

## Cualificación exame

### Os alumnos farán unha

proba escrita ao finalizar o cuadrimestre de acordo ao calendario oficial.

O exame constará dunha parte tipo test (40% da nota final) e unha parte práctica (60%) de realización de pequenos programas. Nesta segunda parte o alumno poderá consultar o manual de Python.

## Oportunidade adiantada de Decembro

- Para a avaliación da oportunidade adiantada aplicaranse os mesmos criterios.

## Alumnos con máis dunha matrícula

-

A avaliación basearase no recollido nesta guía. A maiores realizarán unha serie de traballos prácticos ademais dos propostos para os alumnos de primeira matrícula.

## \* Normativa de avaliación, revisión e

reclamación das cualificacións dos estudos de grao e máster

universitario, aprobadas polo Consello de Goberno da Universidade dá

Coruña o 19 de decembro de 2013.

## Sources of information

Sources of information	
<b>Basic</b>	<ul style="list-style-type: none"><li>- Jesús J. García Molina, Francisco J. Montoya Dato, José L. Fernández Alemán, M<sup>a</sup> José Majado Rosales (2005). Una introducción a la programación : un enfoque algorítmico. Thomson</li><li>- Luis Joyanes Aguilar (2008). Fundamentos de programación : algoritmos, estructuras de datos y objetos. McGraw Hill</li><li>- Raúl González Duque (). Python PARA TODOS. <a href="http://edge.launchpad.net/improve-python-spanish-doc/0.4/0.4.0/+download/Python%20para%20todos.pdf">http://edge.launchpad.net/improve-python-spanish-doc/0.4/0.4.0/+download/Python%20para%20todos.pdf</a></li><li>- Mark Lutz (2013). Learning Python, Fifth Edition. O'Reilly Media, Inc</li><li>- Vernon L Ceder (2010). The quick Python book. Greenwich : Manning</li><li>- Ljubomir Perkovic (2015). Introduction to Computing Using Python: An Application Development Focus, 2nd Edition. Wiley</li></ul>
<b>Complementary</b>	<ul style="list-style-type: none"><li>- Bill Lubanovic (2014). Introducing Python: Modern Computing in Simple Packages. O'Reilly Media</li><li>- Mitchell L Model (2009). Bioinformatics Programming Using Python. O'Reilly Media</li></ul>

## Recommendations

Subjects that it is recommended to have taken before



Subjects that are recommended to be taken simultaneously
Subjects that continue the syllabus
Introduction to databases/614522002
Data structures and algorithmics for biological sequences/614522013
Other comments

(\*)The teaching guide is the document in which the URV publishes the information about all its courses. It is a public document and cannot be modified. Only in exceptional cases can it be revised by the competent agent or duly revised so that it is in line with current legislation.