UNIVERSIDADE DA CORUÑA

| Teaching Guide | | | | |
|---|---|---|---|---|
| **Identifying Data** | | | | **2024/25** |
| **Subject (*)** | Programming I | | **Code** | 614G03006 |
| **Study programme** | Grao en Intelixencia Artificial | | | |
| Descriptors | | | | |
| **Cycle** | **Period** | **Year** | **Type** | **Credits** |
| Graduate | 1st four-month period | First | Basic training | 6 |
| **Language** | SpanishGalician | | | |
| **Teaching method** | Face-to-face | | | |
| **Prerequisites** | | | | |
| **Department** | Ciencias da Computación e Tecnoloxías da Información | | | |
| **Coordinador** | Rabuñal Dopico, Juan Ramon | **E-mail** | juan.rabunal@udc.es | |
| **Lecturers** | Alvarellos González, Alberto José<br>Cedrón Santaeufemia, Francisco Abel<br>Martinez Perez, Maria<br>Rabuñal Dopico, Juan Ramon | **E-mail** | alberto.alvarellos@udc.es<br>francisco.cedron@udc.es<br>maria.martinez@udc.es<br>juan.rabunal@udc.es | |
| **Web** | campusvirtual.udc.gal | | | |
| **General description** | Esta materia é unha introdución á programación, na que se ve como resolver problemas nunha linguaxe estruturada. Nela axúdase ao alumno para comprender os tipos e estruturas de datos básicos, ao mesmo tempo que sentan as bases para deseñar correctamente un algoritmo. E para asentar os coñecementos fundamentais da programación de forma máis rápida e óptima é necesario utilizar unha linguaxe que permita a posta en práctica dos coñecementos adquiridos e sirva de base para o bo desenvolvemento dun programador informático; utilizarase a Linguaxe de programación Python, tanto para as prácticas como para os exemplos teóricos. | | | |

| **Study programme competences / results** | |
|---|---|
| **Code** | **Study programme competences / results** |
| A2 | Capacidad para resolver problemas de inteligencia artificial que precisen algoritmos, aplicando correctamente metodologías de desarrollo software y diseño centrado en usuario/a. |
| A3 | Capacidad para comprender y dominar los conceptos básicos de lógica, gramáticas y lenguajes formales para analizar y mejorar las soluciones basadas en inteligencia artificial. |
| B2 | Que el alumnado sepa aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posea las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio. |
| B3 | Que el alumnado tenga 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. |
| B4 | Que el alumnado pueda transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado. |
| B5 | Que el alumnado haya desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía. |
| B6 | Capacidad para concebir, redactar, organizar, planificar, y desarrollar modelos, aplicaciones y servicios en el ámbito de la inteligencia artificial, identificando objetivos, prioridades, plazos recursos y riesgos, y controlando los procesos establecidos. |
| B7 | Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. |
| B8 | Capacidad para diseñar y crear modelos y soluciones de calidad basadas en Inteligencia Artificial que sean eficientes, robustas, transparentes y responsables. |
| B9 | Capacidad para seleccionar y justificar los métodos y técnicas adecuadas para resolver un problema concreto, o para desarrollar y proponer nuevos métodos basados en inteligencia artificial. |
| C2 | Capacidad de trabajo en equipo, en entornos interdisciplinares y gestionando conflictos. |
| C3 | Capacidad para crear nuevos modelos y soluciones de forma autónoma y creativa, adaptándose a nuevas situaciones. Iniciativa y espíritu emprendedor. |
| C6 | Capacidad para integrar aspectos jurídicos, sociales, ambientales y económicos inherentes a la inteligencia artificial, analizando sus impactos, y comprometiéndose con la búsqueda de soluciones compatibles con un desarrollo sostenible. |

| Learning outcomes | | | |
|---|---|---|---|
| Learning outcomes | Study programme competences / results | | |
| Coñecer e comprender a importancia dos obxectivos da programación. Coñecer os aspectos xerais sobre as linguaxes e paradigmas da programación. Coñecer pseudocódigo e a sintaxe da linguaxe Phyton utilizado para describir algoritmos e programas. Coñecer os pasos para a realización dun programa e os seus principais compoñentes. Coñecer os tipos de datos básicos usando a Linguaxe Phyton. Coñecer as estruturas de control da programación estruturada e as diferenzas entre elas. Coñecer todos os aspectos relacionados coa realización de funcións. Levar a cabo o proceso que permite, desde a abstracción, implementar código de alta calidade. Aplicar programación modular para resolver problemas específicos no ámbito de IA. Comprender a sintaxe e semántica da linguaxe de programación. | A2<br>A3 | B2<br>B3<br>B5<br>B7<br>B8<br>B9 | C3<br>C6 |
| Ser capaz de realizar el seguimiento de un algoritmo (en pseudocódigo) o programa (en Lenguaje Phyton ), explicar qué realiza, y encontrar posibles errores. Ser capaz de resolver pequeños algoritmos y programas. A partir del planteamiento de un problema de pequeña-mediana envergadura saber realizar el programa para resolverlo: teniendo en cuenta los objetivos de la programación estructurada. Realizar la descomposición adecuada implementando las funciones y procedimientos necesarios correctamente. Emplear un estilo de programación apropiado: saber hacer buen uso de identificadores, comentarios justos, saber establecer precondiciones y postcondiciones, saber realizar un buen diseño de las interfaces de procedimientos y funciones, saber elegir y utilizar los tipos y estructuras de datos adecuados, saber elegir y utilizar las estructuras de control convenientes. Saber hacer buen conocimiento de la parte del lenguaje que se explique. Adquirir competencias para resolver problemas de forma metodológica y práctica. Identificar y tener la capacidad para seleccionar en un entorno práctico las principales librerías, coger experiencia en la utilización de librerías, que será fundamental en el desarrollo de la profesión donde se necesite programación para la IA y Ciencia de Datos. Analizar las alternativas para afrontar un problema e identificar qué aspectos pueden abordarse, habilidad muy importante en la IA. Manejar técnicas y herramientas de prueba para asegurar la calidad de los resultados | A2<br>A3 | B2<br>B3<br>B4<br>B5<br>B6<br>B7<br>B8<br>B9 | C2<br>C3 |

| Contents | |
|---|---|
| Topic | Sub-topic |

| | |
|---|---|
| 1 CONCEPTOS BÁSICOS. PARADIGMA IMPERATIVO | 1.1 Algoritmos<br>1.1.1 Representación de algoritmos<br><br>1.2 Programas<br>1.2.1 Tipos de programas<br><br>1.3 Linguaxes de programación<br>1.3.1 Unha visión histórica<br>1.3.2 Clasificación dás linguaxes<br>1.3.3 Instrucións máis importantes<br>1.3.4 Propiedades das linguaxes<br><br>1.4 Tradutores<br><br>1.5. Descripción das linguaxes<br><br>1.6 Estrutura dun programa<br><br>1.7 Elementos dun programa<br>1.7.1 Símbolos  predefinidos<br>1.7.2 Símbolos especiais<br>1.7.3 Identificadores<br>1.7.4 Etiquetas<br>1.7.5 Comentarios<br>1.7.6 Directivas<br>1.7.7 Constantes<br>1.7.8 Números<br>1.7.9 Cadeas de caracteres<br>1.7.10 Variables: Declaración e inicialización<br><br>1.8 Saída e Entrada<br>1.8.1 Sentenzas de saída<br>1.8.2 Sentenzas de entrada<br><br>1.9 Tipos de datos e operadores<br>1.9.1 Tipos de datos<br>1.9.2 Operadores<br>1.9.3 Expresións<br><br>1.10 Importar e usar librerías<br><br>1.11 Depuración de programas |

| | |
|---|---|
| 2 SENTENZAS DE CONTROL | 2.1 Secuencial<br><br>2.2 Condicional<br>2.2.1 A sentenza condicional simple<br>2.2.2 A sentenza condicional múltiple<br><br>2.3 Repetitiva<br>2.3.1 Introdución<br>2.3.2 Variables asociadas aos bucles<br>2.3.3 Funcionamento dos diferentes tipos de bucles<br>2.3.4 Bucle While<br>2.3.5 Bucle FOR<br>2.3.6 Equivalencia entre bucles<br>2.3.7 Erros nos bucles<br>2.3.8 Deseño de bucles |
| 3 ARQUITECTURA DUN PROGRAMA | 3.1 Funcións<br>3.1.1 Tipos de funcions<br>3.1.2 Función como argumentos<br><br>3.3 Recursividade<br>3.3.1 Natureza da recursividade<br>3.3.2 Recursión infinita |
| 4 ESTRUCTURAS SIMPLES DE DATOS | 4.1 Lista e Tuplas<br>4.1.1 Tipo de datos<br>4.1.2 Operacións con Listas e Tuplas<br><br>4.2 Diccionarios<br><br>4.3 Cadeas de caracteres<br>4.3.1 Cadeas de lonxitude variable |
| 5 ENTRADA / SAÍDA | 5.1 Ficheiros<br>5.2 Tipos de ficheiros |
| 6 TESTEO E PROBAS DE PROGRAMAS | 6.1 Depuración de programas<br>6.2 Detección de erros<br>6.2.1 En tempo de compilación<br>6.2.2 En tempo de execución |

| Planning | | | | |
|---|---|---|---|---|
| **Methodologies / tests** | **Competencies / Results** | **Teaching hours (in-person & virtual)** | **Student?s personal work hours** | **Total hours** |
| Laboratory practice | A2 A3 B2 B3 B4 B5 B6 B7 B8 B9 C2 C3 C6 | 28 | 60 | 88 |
| Objective test | A2 A3 B4 B5 B6 B7 B8 B9 | 2 | 2 | 4 |
| Guest lecture / keynote speech | A2 A3 B2 B3 B5 B7 B8 B9 C3 C6 | 28 | 28 | 56 |
| Personalized attention | | 2 | 0 | 2 |
| **(*)The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.** | | | | |

| Methodologies | |
|---|---|
| **Methodologies** | **Description** |
| Laboratory practice | Nas sesións de prácticas o alumno diseñará o pseudocódigo do problema a resolver (papel ou ordenador) para despois codificalo en Linguaxe Phyton, compilalo, executalo e comprobar o seu nivel de corrección.<br><br>Os enunciados dos programas proporcionarase coa suficiente antelación para que os alumnos poidan aproveitar mellor o seu tempo.<br><br>É misión do profesor supervisar o código xerado polo alumno para resolver dúbidas, corrixir malos estilos de programación e corrixir erros, contando con que o profesor non é un compilador que busca erros. |
| Objective test | Para evaluar el aprendizaje, se realizará una prueba escrita que constará de varios ejercicios a realizar en lenguaje de programación Python. Se realizará en las fechas establecidas por la Xunta de Facultade. |
| Guest lecture / keynote speech | Nas sesións de teoría, o profesor describe os obxectivos e os contidos da materia, para dar unha visión particular do tema para tratar e relacionalo con outros dentro da materia<br><br>Despois desenvólvese o tema correspondente na forma de sesión maxistral, axudándose das ferramentas técnicas dispoñibles, facendo fincapé en certas cuestións nas que o alumno debe profundar no seu autoaprendizaje.<br><br>O obxectivo é que o alumno aprenda a algoritmizar, utilizar as estruturas básicas de datos e resolver sinxelos problemas de programación. Utilizarase como linguaxe de codificación Phyton<br><br>As sesión maxistrais poden ser presenciais ou a través de plataformas informáticas como TEAMS (en casos excepcionais). Tamén se poden incluír vídeos explicativos de diferentes partes dos contidos teóricos |

| Personalized attention | |
|---|---|
| **Methodologies** | **Description** |
| Guest lecture / keynote speech<br>Laboratory practice<br>Objective test | Tanto nas sesións maxistrais como nos laboratorios de prácticas levará unha atención personalizada do alumno, en distintos niveis segundo sexa o tipo de clase, detectando o nivel de asimilación e comprensión dos temas explicados e as prácticas requiridas a implantar.<br><br>Nas sesións de práctica é onde se pode chegar máis ao alumno para coñecer as lagoas que ten, e indicarlle o camiño para cubrilas.<br><br>Os alumnos que teñan matrícula a tempo parcial deben, ao comezo do curso, falar co/os profesores encargados do seu grupo. |

| Assessment | | | |
|---|---|---|---|
| **Methodologies** | **Competencies / Results** | **Description** | **Qualification** |
| Laboratory practice | A2 A3 B2 B3 B4 B5 B6 B7 B8 B9 C2 C3 C6 | Durante as últimas semanas con prácticas do curso realizarase unha proba no laboratorio usando ordenadores que terá un valor máximo de 4 puntos sobre a nota total do curso. Será necesario que o programa para realizar polo alumno no laboratorio compile e execute de forma correcta e completa. | 40 |
| Objective test | A2 A3 B4 B5 B6 B7 B8 B9 | O EXAME FINAL, tanto na primeira convocatoria (xaneiro) como na segunda oportunidade (xuño/xullo) constará de varias preguntas ou exercicios que o alumno terá que desenvolver en código Phyton, e terá un valor de 6 puntos. | 60 |

| Assessment comments |
|---|

A nota final virá dada pola nota obtida por AVALIACIÓN CONTINUA e a obtida no EXAME FINAL. O Exame Final constará de varias preguntas e problemas a codificar na linguaxe de programación empregado nas sesións prácticas. Para poder superar a materia será obrigatorio obter polo menos o 30% da cualificación do EXAME FINAL e ademais, alcanzar polo menos un 5 (sobre 10 puntos) tras sumar a nota do EXAME FINAL, coa cualificación obtida de AVALIACIÓN CONTINUA.
Non Presentado: os estudantes que non participen no EXAME FINAL terán a cualificación de "Non Presentado".
Para os alumnos que teñan matrícula a tempo parcial a asistencia as clases non é obrigatoria, pero si a asistencia a proba no laboratorio na data establecida para a avaliación da parte práctica de laboratorio.
Durante a realización da proba, en calquera de ambas as oportunidades, agás que se indique o contrario, está prohibido o uso de calquera dispositivo con acceso a Internet. Se durante a realización da proba, hai indicio do uso non autorizado deses dispositivos, o estudante será expulsado da aula, e procederase segundo a Lei 3/2022, do 24 de febreiro, de convivencia universitaria e o regulamento disciplinar do estudantado da UDC.
A realización fraudulenta das probas e/o actividades implicará directamente a cualificación de suspenso ("0") na materia na convocatoria correspondente, invalidando calquera cualificación obtida en todas as actividades para a seguinte oportunidade, de existir, dentro do mesmo curso académico. Considérase fraudulenta a realización das actividades, propostas a ser completadas presencialmente na aula, que se fagan desde fóra da aula, procedendo segundo a Lei 3/2022, do 24 de febreiro, de convivencia universitaria e o regulamento disciplinar do estudantado da UDC.

| Sources of information | |
|---|---|
| Basic | - Charles Russell Severance (Autor), Fernando Tardio Muniz (Traductor) (2015). Python para informaticos: Explorando la informacion.<br>- Charles Russell Severance (2016). Python for Everybody: Exploring Data in Python 3. Open Textbook Library<br>- Alberto Cuevas Alvarez (2016). Python 3. Curso Práctico. Editorial RA-MA<br>- Mark Lutz (2013). Learning Python. Quinta edición . O?Reilly Media Inc<br>- Arturo Montejo Ráez, Salud María Jiménez Zafra (2019). Curso de Programación Python. Editorial Anaya |
| Complementary | - Mark Summerfield (2010). Python 3. Editorial Anaya<br>- Sébastien Chazallet (2016). Python 3. Los fundamentos del lenguaje - 2ª edición. Ediciones-ENI<br>- Raúl González Duque (2008). Python para todos.<br>- John V. Guttag (2013). Introduction to Computation and Programming Using Python. The MIT Press |

| Recommendations |
|---|
| **Subjects that it is recommended to have taken before** |
| |
| **Subjects that are recommended to be taken simultaneously** |
| Introduction to Computers/614G03012 |
| **Subjects that continue the syllabus** |
| Programming II/614G03007 |
| **Other comments** |

O alumno debe ter en conta que debe realizar un labor autodidacta moi importante, seguindo o seguinte esquema: Ler, atender, comprender, preguntar, estudar e practicar:- Ler: Lea o tema para tratar antes de asistir ás sesións teóricas. É MOI IMPORTANTE!- Atender: Atenda en clase, non só estea de corpo presente.- Comprender: Comprenda o que se lle di nas sesións de teoría, e se non pregunte.- Preguntar: Pregunte todo o que non comprenda, non quede con dúbidas.- Estudar: Estude despois das sesións, para reter o comprendido. - Practicar: Faga moitos programas, os que se lle pidan, suxiran, e outros pola súa conta, tanto en papel como no computador. Programación é unha materia que non se pode aprender estudando en dous días. O alumno debe ir madurando os conceptos, facer sobre o papel e na máquina moitos programas, aprendendo tamén dos erros ao realizalos.É unha materia que, por medio do sistema de avaliación continua, pódese aprobar sen máis que seguir, de forma activa, o ritmo das distintas sesións teóricas e prácticas. Debe facer caso ás indicacións particulares de reforzo de estudo que lle sinale o profesor. Perspectiva de xénero: tal e como se recolle nas competencias transversais do título (C4), fomentarase o desenvolvemento dunha cidadanía crítica, aberta e respectuosa coa diversidade na nosa sociedade, salientando a igualdade de dereitos do alumnado sen discriminación por cuestión de xénero ou condición sexual. Empregarase unha linguaxe inclusiva no material e no desenvolvemento das sesións. Traballarase para identificar e modificar prexuizos e actitudes sexistas e influirase na contorna para modificalos e fomentar valores de respecto e igualdade.

**(*)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.**