Guía Docente				
Datos Identificativos				2019/20
Asignatura (*)	Programación I		Código	614G01001
Titulación	Grao en Enxeñaría Informática			

Descriptores				
Ciclo	Período	Curso	Tipo	Créditos
Grao	1º cuadrimestre	Primeiro	Formación básica	6

Idioma	CastelánInglés		
Modalidade docente	Presencial		
Prerrequisitos			
Departamento	Ciencias da Computación e Tecnoloxías da InformaciónComputación		
Coordinación	Rabuñal Dopico, Juan Ramon	Correo electrónico	juan.rabunal@udc.es
Profesorado	Arcay Varela, Bernardino Boveda alvarez, Maria del Carmen Castro Martinez, Alfonso Cedrón Santaeufemia, Francisco Abel Martinez Perez, Maria Munteanu , Cristian Robert Rabuñal Dopico, Juan Ramon	Correo electrónico	bernardino.arcay@udc.es carmen.boveda@udc.es alfonso.castro@udc.es francisco.cedron@udc.es maria.martinez@udc.es c.munteanu@udc.es juan.rabunal@udc.es
Web	moodle.udc.es/		
Descrición xeral	Esta materia é unha introdución á programación, na que se ve cómo resolver problemas nunha linguaxe estructurada. Nela axúdase ao alumno a comprender os tipos e estruturas de datos básicos, ao mesmo tempo que se sentan as bases para deseñar correctamente un algoritmo. E para asentar os coñecementos fundamentáis da programación de forma máis rápida e óptima é necesario empregar unha linguaxe que permita a posta en práctica dos coñecementos adquiridos e sirva de base para o bó desenvolvemento dun programador informático; utilizarase a Linguaxe C, tanto para as prácticas como para os exemplos teóricos.		

Competencias / Resultados do título	
Código	Competencias / Resultados do título
A4	Coñecementos básicos sobre o uso e a programación dos ordenadores, sistemas operativos, bases de datos e programas informáticos con aplicación na enxeñaría.
A5	Coñecemento da estrutura, organización, funcionamento e interconexión dos sistemas informáticos, os fundamentos da súa programación e a súa aplicación para a resolución de problemas propios da enxeñaría.
B1	Capacidade de resolución de problemas
B3	Capacidade de análise e síntese
B4	Capacidade para organizar e planificar
C3	Utilizar as ferramentas básicas das tecnoloxías da información e as comunicacións (TIC) necesarias para o exercicio da súa profesión e para a aprendizaxe ao longo da súa vida.
C6	Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben enfrontarse.
C7	Asumir como profesional e cidadán a importancia da aprendizaxe ao longo da vida.

Resultados da aprendizaxe			
Resultados de aprendizaxe	Competencias / Resultados do título		
Coñecer e comprender a importancia dos obxectivos da programación. Coñecer os aspectos xerais sobre as linguaxes e paradigmas da programación. Coñecer pseudocódigo e a sintaxis da linguaxe C 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 C. Coñecer as estruturas de control da programación estructurada e as diferenzas entre elas. Coñecer todos os aspectos relacionados coa realización de funcións e procedementos.	A4 A5	B1 B3 B4	

Coñecer e comprender a importancia dos obxectivos da programación. Coñecer os aspectos xerais sobre as linguaxes e paradigmas da programación. Coñecer pseudocódigo e a sintaxis da linguaxe C 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 C. Coñecer as estruturas de control da programación estructurada e as diferenzas entre elas. Coñecer todos os aspectos relacionados coa realización de funcións e procedementos.	A4 A5	B1 B3 B4	
Ser capaz de realizar o seguimiento dun algoritmo (en pseudocódigo) ou programa (en linguaxe C), explicar que realiza, e atopar posibles erros. Ser capaz de resolver pequenos algoritmos e programas. A partir da formulación dun problema de pequena-mediana envergadura saber realizar o programa para resolvelo: tendo en conta os obxectivos da programación. Realizar a descomposición adecuada implementando as funcións e procedementos necesarios correctamente. Empregar un estilo de programación apropiado: saber facer bo uso de identificadores, comentarios xustos, saber establecer precondiciones e postcondiciones, saber realizar un bo deseño das interfaces de procedementos e funcións, saber elixir e utilizar os tipos e estruturas de datos adecuados, saber elixir e utilizar as estruturas de control convenientes. Saber facer bo coñecemento da parte da linguaxe que se explique.	A4 A5	B1 B3 B4	C3 C6 C7
Ser capaz de realizar o seguimiento dun algoritmo (en pseudocódigo) ou programa (en linguaxe C), explicar que realiza, e atopar posibles erros. Ser capaz de resolver pequenos algoritmos e programas. A partir da formulación dun problema de pequena-mediana envergadura saber realizar o programa para resolvelo: tendo en conta os obxectivos da programación. Realizar a descomposición adecuada implementando as funcións e procedementos necesarios correctamente. Empregar un estilo de programación apropiado: saber facer bo uso de identificadores, comentarios xustos, saber establecer precondiciones e postcondiciones, saber realizar un bo deseño das interfaces de procedementos e funcións, saber elixir e utilizar os tipos e estruturas de datos adecuados, saber elixir e utilizar as estruturas de control convenientes. Saber facer bo coñecemento da parte da linguaxe que se explique.		B1 B3 B4	C3 C6 C7
Aprendizaxe autónoma. Planificación das actividades a desenvolver. Capacidade de abstracción. Toma de decisións. Capacidade de iniciativa e participación.		B3 B4	C3 C6 C7
Aprendizaxe autónoma. Planificación das actividades a desenvolver. Capacidade de abstracción. Toma de decisións. Capacidade de iniciativa e participación.		B3 B4	C3 C6 C7

Contidos	
Temas	Subtemas

1 CONCEPTOS BÁSICOS	1.1 Algoritmos 1.1.1 Representación de algoritmos 1.2 Programas 1.2.1 Tipos de programas 1.3 Linguaxes de programación 1.3.1 Unha visión histórica 1.3.2 Clasificación das linguaxes 1.3.3 Instrucións máis importantes 1.3.4 Propiedades das linguaxes 1.4 Tradutores e Compiladores 1.5 Estrutura dun programa 1.6 Elementos dun programa 1.6.1 Símbolos predefinidos 1.6.2 Símbolos especiais 1.6.3 Identificadores 1.6.4 Etiquetas 1.6.5 Comentarios 1.6.6 Directivas 1.6.7 Constantes 1.6.8 Números 1.6.9 Cadeas de carácteres 1.6.10 Variables: Declaración e iniciación 1.6.11 Variables: Dirección de Memoria 1.7 Saída e Entrada 1.7.1 Sentenzas de saída 1.7.2 Sentenzas de entrada 1.8 Tipos de datos e operadores 1.8.1 Tipos de datos 1.8.2 Operadores 1.8.3 Expresións

2 SENTENZAS DE CONTROL	2.1 Secuencial 2.2 Alternativa 2.2.1 A sentenza condicional simple 2.2.2 A sentenza condicional múltiple 2.3 Repetitiva 2.3.1 Introdución 2.3.2 Variables asociadas aos bucles 2.3.3 Funcionamento dos diferentes tipos de bucles 2.3.4 Bucle FOR 2.3.5 Equivalencia entre bucles 2.3.6 Erros nos bucles 2.3.7 Deseño de bucles
3 ARQUITECTURA DUN PROGRAMA	3.1 Funcións e Procedementos 3.1.1 Tipos de funciones y procedimientos 3.1.2 Parámetros por valor e referencia 3.1.3 Parámetros protexidos 3.1.4 A pila de activación de procedementos e funcións 3.1.5 Variables globais e locais: Alcance 3.1.6 Efectos laterais 3.2 Recursividade 3.2.1 Natureza da recursividade 3.2.2 Recursión infinita
4 ESTRUTURAS SIMPLES DE DATOS	4.1 Arrays e Matrices 4.1.1 Tipo de dato ARRAY 4.1.2 Declaración dun Array 4.1.3 Arrays de máis dunha dimensión 4.1.4 Operacións con Arrays e Matrices 4.2 Rexistros 4.2.1 Tipo de dato rexistro 4.2.2 Operacións con rexistros 4.3 Cadeas de caracteres 4.3.1 Cadeas de lonxitude fixa 4.3.2 Cadeas de lonxitude variable 4.4 Operacións básicas sobre Arrays 4.4.1 Búsqueda 4.4.2 Ordenación
5 ENTRADA / SAIDA	5.1 Ficheiros 5.2 Tipos 5.3 Operaciones e modos de acceso 5.4 Funcións e procedementos predefinidos específicos

Planificación

Metodoloxías / probas	Competencias / Resultados	Horas lectivas (presenciais e virtuais)	Horas traballo autónomo	Horas totais
Sesión maxistral	A4 A5 B1 B3 C6 C7	30	30	60
Prácticas de laboratorio	A4 A5 B1 B3 B4 C3 C6 C7	20	50	70
Seminario	B4 C3 C6	8	10	18
Atención personalizada		2	0	2
*Os datos que aparecen na táboa de planificación son de carácter orientativo, considerando a heteroxeneidade do alumnado				

Metodoloxías	
Metodoloxías	**Descrición**
Sesión maxistral	Nas sesións de teoría, o profesor describe os obxectivos e os contidos da materia, para dar unha visión particular do tema a tratar e relacionalo con outros dentro da asignatura 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 autoaprendizaxe. 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 o C
Prácticas de laboratorio	Nas sesións de prácticas o alumno realizará programas en papel para despois codificalo en Linguaxe C, compilalo, executalo e comprobar o seu nivel de corrección. Os enunciados dos programas proporcionaranse coa suficiente antelación para que os alumnos poidan aproveitar mellor o seu tempo. 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.
Seminario	Nas sesións de seminario realizaranse exercicios e prácticas coa finalidade de detectar nos alumnos lagoas de coñecemento na materia impartida ata ese momento, e dar as explicacións e/ou referencias necesarias para enmendalas

Atención personalizada	
Metodoloxías	**Descrición**
Prácticas de laboratorio Seminario Sesión maxistral	Tanto nas sesións maxistrales como nos laboratorios de prácticas e nas sesións de seminario levarase 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. Nas sesións de seminario é onde se pode chegar máis ao alumno para coñecer as lagoas que presente e indicarlle o camiño para cubrilas. Os alumnos que teñan matrícula a tempo parcial deben falar, o comezo do curso, con o/os profesores encargados do seu grupo.

Avaliación			
Metodoloxías	**Competencias / Resultados**	**Descrición**	**Cualificación**
Prácticas de laboratorio	A4 A5 B1 B3 B4 C3 C6 C7	Durante as últimas semanas con prácticas do curso realizarase unha proba no laboratorio usando ordenadores que terá un valor máximo de 3 puntos sobre a nota total do curso (puntuada sobre 10). Será necesario que o programa a realizar polo alumno no laboratorio compile e execute de forma correcta e completa.	30

Sesión maxistral	A4 A5 B1 B3 C6 C7	A nota da asignatura será a suma do obtido na Avaliación Continua (durante as 15 semanas do periodo lectivo correspondente á asignatura) e o obtido no Exame Final. A nota de AVALIACIÓN CONTINUA, valorada en 4 puntos, divídese en duas partes: 1.- Á metade do curso realizarase unha proba escrita que valerá 1 punto. 2.- Nas últimas semanas de prácticas do curso realízarase unha proba no laboratorio utilizando ordenadores que valerá un máximo de 3 puntos. O EXAME FINAL constará de varios exercicios que o alumno terá que desenvolver en código C, e terá un valor de 6 puntos. O exame oficial, tanto na primeira (xaneiro) como na segunda oportunidade (xullo) constará de varios problemas a desenvolver en código C. Devandito Exame Final terá un valor máximo de 6 puntos, que se sumarán ao obtido na Avaliación Continua.	70

Observacións avaliación
A nota final virá dada pola nota obtida por AVALIACIÓN CONTINUA e a obtida no EXAME FINAL. O Exame Final constará de tres problemas a codificar na linguaxe empregada nas sesións prácticas.

Fontes de información	
Bibliografía básica	- Kernighan, Brian W. Englewood Cliffs (1988). The C Programming Language. New Jersey. Prentice Hall - K.N. King (2008). C programming. A modern Approach. Second Edition.. - James L. Antonakos , Kenneth C. Mansfield (2004). Programación estructurada en C. Madrid. Prentice-Hall - Luis Joyanes Aguilar, Ignacio Zahonero Martínez (2005). Programación en C metodología, algoritmos y estructura de datos. Madrid. McGraw-Hill - José R. García-Bermejo Giner (2008). Programación estructurada en C. Pearson - Luis Joyanes Aguilar (2011). Fundamentos de programación : algoritmos, estructuras de datos y objetos. Madrid. McGraw-Hill
Bibliografía complementaria	- Gabriela Márquez, Sonia Osorio, Noemí Olvera (2011). Introducción a la Programación Estructurada en C. Pearson - Andrés Marzal, Isabel García (2017). Introducción a la Programación con C. Publicacions de la Universitat Jaume I. Servei de Comunicació i Publicacions - Luis Joyanes Aguilar (2002). Programación en C. libro de problemas. Madrid. McGraw-Hill

Recomendacións
Materias que se recomenda ter cursado previamente
Materias que se recomenda cursar simultaneamente
Informática Básica/614G01002
Materias que continúan o temario
Programación II/614G01006
Observacións
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 a 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 si non pregunte. Preguntar: Pregunte todo o que non comprenda, non quede con dúbidas. Estudar: Estude logo 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 ordenador.   Programación é unha asignatura 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 asignatura 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.

(*)A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente dacordo coa normativa vixente que establece o proceso de elaboración de guías