

Plan 460 GRADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

Asignatura 45004 PROGRAMACION

Tipo de asignatura (básica, obligatoria u optativa)

Formación básica

Créditos ECTS

6 ECTS

Competencias que contribuye a desarrollar

- GB1. Capacidad de razonamiento, análisis y síntesis.
- GB2. Capacidad para relacionar conceptos y adquirir una visión integrada, evitando enfoques fragmentarios.
- GB3. Capacidad de toma de decisiones en la resolución de problemas básicos de ingeniería de telecomunicación, así como identificación y formulación de los mismos.
- GB4. Capacidad para trabajar en grupo, participando de forma activa, colaborando con sus compañeros y trabajando de forma orientada al resultado conjunto, y en un entorno multilingüe.
- GB5. Conocimiento de materias básicas, científicas y tecnologías, que le capacite para el aprendizaje de nuevos métodos y tecnologías.
- GBE3. Capacidad para resolver problemas con iniciativa, creatividad y razonamiento crítico.
- GC1. Capacidad de organización, planificación y gestión del tiempo.
- GC2. Capacidad para comunicar, tanto por escrito como de forma oral, conocimientos, procedimientos, resultados e ideas relacionadas con las telecomunicaciones y la electrónica.
- GC3. Trabajar en cualquier contexto, individual o en grupo, de aprendizaje o profesional, local o internacional, desde el respeto a los derechos fundamentales, de igualdad de sexo, raza o religión y los principios de accesibilidad universal, así como la cultura de paz.

Objetivos/Resultados de aprendizaje

Al finalizar la asignatura el alumno deberá ser capaz de:

- Conocer los conceptos relacionados con la programación.
- Aplicar las técnicas y procedimientos de una metodología de programación de un sistema software.
- Codificar y probar dicho sistema, aplicando técnicas de programación orientada a procesos y a datos.
- Codificar, poner a punto y ejecutar programas sencillos en lenguaje C.
- Diseñar algoritmos sencillos basados en los esquemas de recorrido y búsqueda.
- Autoevaluar el trabajo realizado e identificar los propios errores y aspectos a mejorar.
- Aprender de forma autónoma
- Trabajar en equipo

Contenidos

5.

Bloques temáticos

Las clases teóricas se articulan en cuatro bloques temáticos.

Los objetivos de aprendizaje describen en detalle todo lo que el alumno va a aprender durante este curso. Es importante que el alumno los tenga presente desde el primer momento, aunque el profesor le irá recordando los objetivos que están implicados en las diferentes actividades del curso.

5.1. Características comunes a todos los bloques

a.

Métodos docentes

- Clase expositiva participativa.
- Estudio de casos en laboratorio.
- Resolución de problemas.
- Aprendizaje cooperativo.
- Aprendizaje basado en proyecto.

b.

Plan de trabajo

Ver Anexo I de guía docente.

c.

Evaluación

- Entregables individuales y de equipo.
- Pruebas de conocimientos y habilidades básicas.
- Evaluación de entregas parciales del Proyecto Global.
- Evaluación de la versión final del Proyecto Global.

d.

Bibliografía básica

- Apuntes de la asignatura.
- Ver Anexo II (bibliografía) de guía docente.

e.

Bibliografía complementaria

- Ver Anexo II (bibliografía) de guía docente.

i.

Recursos necesarios

- Aula con proyector multimedia y pizarra para sesiones de discusión.
- Laboratorio de prácticas, con un ordenador por alumno, para las sesiones de laboratorio. Cada ordenador debe contar con los entornos de desarrollo para el lenguaje C que se van a utilizar.
 - Plataforma educativa para publicar material didáctico, guías de ejercicios, soluciones, tareas, etc.
 - Foro de discusión en línea para atender preguntas y discutir aspectos relacionados con el aprendizaje de la asignatura.
 - Acceso al material bibliográfico recomendado.

5.3. Desglose por bloques

Bloque 1:

Conceptos Básicos

Carga de trabajo en créditos ECTS:

1.6

a.

Contextualización y justificación

En este primer bloque se introducen los conceptos básicos sobre los diferentes métodos para el desarrollo de aplicaciones informáticas, se presenta C, el lenguaje de programación que se utilizará para la implantación de los programas desarrollados, y se proporcionan los elementos básicos de la programación estructurada. El bloque se completa con casos de estudio y con la aplicación al proyecto global de los métodos aprendidos.

b.

Objetivos de aprendizaje

Al final de este bloque el alumno será capaz de:

- En relación con los tipos de datos elementales y sus operaciones (enteros, caracteres y reales):
 - Escribir los tipos de datos elementales y las operaciones que actúan sobre ellos.
 - Escribir la declaración de datos de cualquiera de los tipos elementales.
 - Indicar el código ASCII de cualquier carácter, con la ayuda de la tabla correspondiente.

- En relación con las sentencias básicas de asignación y de entrada/salida:
 - Describir el funcionamiento de las sentencias básicas.
 - Predecir el resultado de una secuencia de sentencias básicas.
 - Codificar una tarea convenientemente especificada, utilizando la secuencia de sentencias básicas adecuada.
- En relación con los archivos:
 - Explicar el concepto de archivo, para qué sirve, y cuáles son las operaciones típicas sobre archivos de texto (crear, abrir, leer, escribir, preguntar por fin de archivo y cerrar).
 - Escribir las sentencias necesarias para realizar las operaciones básicas con un archivo de texto.
 - Escribir las sentencias necesarias para determinar el tipo de error que se ha producido al realizar una operación con un archivo de texto.
- En relación con las funciones:
 - Definir los conceptos: cabecera de función, parámetros formales, variables locales, resultado de la función, activación de la función, parámetros reales, paso de parámetros.
 - Describir la diferencia entre paso de parámetros por valor o por referencia.
 - Codificar convenientemente una llamada a función, pasando correctamente los parámetros.
 - Codificar en forma de función una tarea convenientemente especificada, estableciendo adecuadamente los parámetros necesarios.
 - Proponer una organización en bloques (funciones) de una aplicación determinada.
- En relación con el entorno de desarrollo:
 - Definir los conceptos de compilación, construcción y ejecución.
 - Realizar las operaciones necesarias para crear/abrir un proyecto, y añadir y eliminar elementos a un proyecto.
 - Realizar las operaciones necesarias para editar, compilar, montar y ejecutar un programa, y localizar las carpetas donde están los archivos generados en cada uno de los pasos.
 - Diseñar archivos makefile.
 - Interpretar adecuadamente los mensajes de error de compilación, y corregir el error de compilación correspondiente.
 - Describir las funcionalidades básicas del depurador.
 - Realizar correctamente las operaciones básicas del depurador (insertar un punto de parada, ejecutar paso a paso y visualizar valores de variables).
 - Identificar y subsanar los errores de ejecución de un programa, utilizando adecuadamente el depurador.

c.
Contenidos

- Introducción a los ordenadores.
- Conceptos básicos del lenguaje C.
- Funciones de Interacción con el usuario y de acceso a archivos de texto.
- Anatomía de un programa.
- Entornos de desarrollo. Depuradores.
- Funciones y procedimientos.
- Programación modular.

Bloque 2:
Sentencias de control

Carga de trabajo en créditos ECTS:
1.2

a.
Contextualización y justificación
El bloque 2 se centra en las sentencias de control. Se presentan las características de C que se utilizarán en esta metodología. El bloque se completa con casos de estudio y con la aplicación al proyecto global de los métodos aprendidos.

b.
Objetivos de aprendizaje

Al final de este bloque el alumno será capaz de:

- En relación con las sentencias de control:
 - Conocer y usar los operadores que se utilizan para construir expresiones lógicas.
 - Usar las sentencias de selección if, if...else para elegir entre varias acciones alternativas.
 - Utilizar la sentencia de selección múltiple switch para escoger entre muchas alternativas de acción.
 - Utilizar las etiquetas case para identificar las acciones alternativas en las sentencias switch.
 - Ejecutar sentencias repetidamente con la sentencia repetitiva for.
 - Usar la sentencia de repetición while para ejecutar sentencias repetidamente.
 - Usar la sentencia de repetición do while para ejecutar sentencias repetidamente.
- En relación con los esquemas algorítmicos (recorrido y búsqueda):
 - Explicar los esquemas de recorrido y búsqueda.
 - Adaptar los esquemas de recorrido y búsqueda a una situación convenientemente especificada, identificando con claridad cada uno de los elementos del esquema.
 - Elegir el esquema adecuado para resolver un problema determinado.
 - Aplicar por iniciativa propia los esquemas algorítmicos estudiados.

c.

Contenidos

- Sentencias de control.
- El preprocesador de C.
- Esquemas algorítmicos.

Bloque 3:

Vectores y Cadenas

Carga de trabajo en créditos ECTS:

1.2

a.

Contextualización y justificación

El bloque 3 se centra en los conceptos de vector y cadena de caracteres, junto con la introducción a los punteros. Se presentan las características de C que se utilizarán en esta metodología. El bloque se completa con casos de estudio y con la aplicación al proyecto global de los métodos aprendidos.

b.

Objetivos de aprendizaje

Al final de este bloque el alumno será capaz de:

- En relación con los vectores:
 - Escribir la declaración de un vector unidimensional.
 - Escribir el código necesario para acceder a un elemento de un vector unidimensional.
 - Escribir la declaración de un vector multidimensional.
 - Escribir el código necesario para acceder a un elemento de un vector multidimensional.
 - En relación a las cadenas de caracteres:
 - Escribir la declaración de una cadena de caracteres.
 - Escribir el código necesario para acceder a un elemento de una cadena de caracteres.
 - Utilizar las funciones de la biblioteca estándar cuyas declaraciones se encuentran en el archivo de cabecera string.h para manipulación de cadenas de caracteres.
 - En relación a los punteros:
 - Comprender el concepto de puntero y sus diferencias con una variable simple.
 - Escribir el código necesario para acceder al dato apuntado por un puntero.
 - Escribir el código necesario para recorrer un vector o cadena de caracteres utilizando un puntero.
 - Implantar el paso de parámetros por dirección utilizando punteros.
-

c.
Contenidos

- Vectores. Esquemas algorítmicos.
- Cadenas de caracteres.
- Punteros.

Bloque 4:
Estructuras de datos

Carga de trabajo en créditos ECTS:
2.0

a.
Contextualización y justificación

El bloque 4 se centra en las estructuras de datos, tanto estáticas como dinámicas, y en la creación e intercambio de datos con archivos binarios. También se introducen los procedimientos para generar estructuras dinámicas de datos. Se presentan las características de C que se utilizarán en esta metodología. El bloque se completa con casos de estudio y con la aplicación al proyecto global de los métodos aprendidos.

b.
Objetivos de aprendizaje

Al final de este bloque el alumno será capaz de:

- En relación con los tipos de datos estructurados:
 - Describir las estructuras de datos fundamentales, y las operaciones típicas sobre ellas.
 - Escribir la declaración de una estructura de datos convenientemente especificada.
 - Escribir el código necesario para acceder a un elemento o conjunto de elementos de una estructura de datos.
 - Elegir la estructura de datos más adecuada para una aplicación determinada.
- En relación con la asignación dinámica de memoria:
 - Asignar y liberar memoria dinámicamente para diferentes tipos de datos.
 - Crear y redimensionar vectores dinámicos.
- En relación con los archivos binarios:
 - Recordar el concepto de archivo, para qué sirve, y cuáles son las operaciones típicas sobre archivos (crear, abrir, leer, escribir, preguntar por fin de archivo y cerrar).
 - Escribir las sentencias necesarias para realizar las operaciones básicas con archivos binarios.
 - Escribir las sentencias necesarias para determinar el tipo de error que se ha producido al realizar una operación con un archivo binario.

c.
Contenidos

- Tipos estructurados de datos en C.
- Archivos binarios.
- Asignación dinámica de memoria.

Principios Metodológicos/Métodos Docentes

- Clase expositiva participativa.
 - Estudio de casos en laboratorio.
 - Resolución de problemas.
 - Aprendizaje cooperativo.
 - Aprendizaje basado en proyecto.
-

Crterios y sistemas de evaluaci3n

A continuaci3n se presenta la tabla que recoge los criterios de evaluaci3n que se aplicarn en la convocatoria ordinaria.

INSTRUMENTO/PROCEDIMIENTO

PESO EN LA NOTA FINAL

OBSERVACIONES

Entregables (individuales y de equipo)

15%

Ser3 necesario realizar adecuadamente y entregar un porcentaje prefijado de los entregables del curso realizados en aula y laboratorio. En caso contrario la calificaci3n final en la asignatura ser3 No Presentado (N.P.).

Conocimientos b3sicos

15%

Cada semana el alumno tendr3 que realizar una prueba objetiva de conocimientos b3sicos sobre el tema de trabajo de dicha semana. Para superar cada prueba dispondr3 de cuatro intentos durante el curso, m3s dos intentos adicionales en las convocatorias de enero y febrero.

Es condici3n necesaria (pero no suficiente) para superar la asignatura que el alumno supere TODOS los conocimientos b3sicos.

Habilidades b3sicas

15%

Al finalizar cada bloque tem3tico el alumno tendr3 que realizar una prueba de habilidades b3sicas. En caso de no superar dicha prueba, tendr3 dos opciones m3s de hacerlo, en las convocatorias de enero y febrero.

Es condici3n necesaria (pero no suficiente) para superar la asignatura que el alumno supere TODAS las habilidades b3sicas.

Proyecto

40%

Realizaci3n de un proyecto integrador de dificultad media, en lenguaje C, que se implementar3 en varias versiones a lo largo del curso. Cada versi3n ser3 una ampliaci3n de la anterior, en la que se incorporarn las nuevas t3cnicas y conocimientos adquiridos. Los alumnos tambi3n tendr3n que coevaluar el proyecto realizado por otros compa1eros. Una vez entregada la 3ltima versi3n del proyecto, el alumno tendr3 que realizar una ampliaci3n individual del mismo. La nota final del proyecto ser3 el producto de la media ponderada de las versiones, multiplicada por la nota de la ampliaci3n individual.

Es condici3n necesaria (pero no suficiente) para superar la asignatura obtener una calificaci3n m3nima de 5/10 puntos en el proyecto, y una calificaci3n m3nima de 0.5/1 en la prueba de ampliaci3n individual del proyecto integrador.

Evaluaci3n

15%

La calificaci3n de este apartado vendr3 dada por las notas obtenidas por el alumno en las actividades de coevaluaci3n y autoevaluaci3n de ejercicios de laboratorio y de las versiones del proyecto integrador.

Si un alumno no alcanza los requisitos m3nimos descritos en la tabla anterior, su calificaci3n final en la asignatura ser3 el m3nimo entre el valor calculado seg3n la ponderaci3n descrita en la tabla y 4.

Recursos de aprendizaje y apoyo tutorial

Ser3n necesarios los siguientes recursos, todos ellos facilitados por el profesor:

- Campus Virtual de la asignatura, que est3 incluido en una plataforma educativa Moodle
- Documentaci3n de apoyo

Tutor3as:

Ver Tutor3as en

<http://www.uva.es/export/sites/uva/2.docencia/2.01.grados/2.01.02.ofertaformativagrados/2.01.02.01.alfabetica/Grado-en-Ingenieria-de-Tecnologias-de-Telecomunicacion/>

Calendario y horario

Se adjunta enlace a la p3gina correspondiente del sitio web de la ETSIT:

http://www.tel.uva.es/bin/horarios1718/Grado_1.pdf

Tabla de Dedicaci3n del Estudiante a la Asignatura/Plan de Trabajo

ACTIVIDADES PRESENCIALES

HORAS

ACTIVIDADES NO PRESENCIALES

HORAS

Clases teórico-prácticas (T/M)

20

Estudio y trabajo autónomo individual

32

Clases prácticas de aula (A)

0

Trabajo autónomo grupal: proyecto

58

Laboratorios (L)

40

Prácticas externas, clínicas o de campo

0

Seminarios (S)

0

Tutorías grupales (TG)

Evaluación

Total presencial

60

Total no presencial

90

Responsable de la docencia (recomendable que se incluya información de contacto y breve CV en el que aparezcan sus líneas de investigación y alguna publicación relevante)

- JOSÉ FERNANDO DÍEZ HIGUERA
DESPACHO: 2D079
TELÉFONO: 983423000 ext. 5562
E-MAIL: josdie@tel.uva.es
- DAVID GONZÁLEZ ORTEGA
DESPACHO: 2D022
TELÉFONO: 983423000 ext. 5552
E-MAIL: davgon@tel.uva.es

Idioma en que se imparte

Castellano