

FUNDAMENTOS DE PROGRAMACIÓN

Curso 2013/2014

(Código: 71901020)

1. PRESENTACIÓN DE LA ASIGNATURA

El objetivo de esta guía es orientar al alumno en el estudio de la asignatura de Fundamentos de Programación. Se recomienda la lectura completa de la guía al comienzo del cuatrimestre para tener una idea completa del temario, las prácticas y el método de evaluación, lo que permitirá al alumno una correcta planificación de su trabajo. Esta asignatura se imparte en el primer cuatrimestre del primer curso, supone una carga docente de 6 créditos y es parte de la materia de fundamentos de programación, con un carácter de formación básica para la titulación de los grados en Ingeniería Informática e Ingeniería de las Tecnologías de la Información.

2. CONTEXTUALIZACIÓN EN EL PLAN DE ESTUDIOS

Esta asignatura, como se define en su título, es un curso de fundamentos de programación y trata de introducir de forma progresiva y sistemática los conceptos para una correcta metodología de programación. Se considera muy importante el acceso de manera natural e inmediata a la programación del computador. Esto permite al alumno comprobar en la práctica que los ejemplos resueltos y los ejercicios propuestos funcionan correctamente. Como primera asignatura de la materia de fundamentos de programación prepara además para los contenidos que se ofrecen en el resto de las asignaturas de esta materia. Con esta asignatura se busca que el estudiante desarrolle las siguientes competencias generales:

Capacidad de análisis y síntesis de problemas. Aplicación de los conocimientos a la práctica.

Resolución de problemas en entornos nuevos. Comunicación y expresión tecnológica

Además, con esta asignatura se trata de alcanzar las siguientes competencias específicas de la materia:

Conocimiento de los fundamentos de la programación de sistemas y su aplicación a la resolución de problemas propios de la ingeniería

Capacidad para identificar y analizar problemas y diseñar, desarrollar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.

Capacidad para emplear metodologías para el desarrollo, evaluación y gestión de aplicaciones y sistemas basados en tecnologías de la información.

3. REQUISITOS PREVIOS REQUERIDOS PARA CURSAR LA ASIGNATURA

Al tratarse de una asignatura de carácter introductorio, no se requiere ningún requisito previo más allá de los conocimientos generales que un alumno debe tener en este nivel de enseñanza y los conocimientos específicos del manejo de un computador personal a nivel de usuario.

4. RESULTADOS DE APRENDIZAJE

El resultado fundamental del aprendizaje de esta asignatura es la adquisición de manera progresiva y sistemática de una correcta metodología para la programación de computadores. Las materias que se aprenden son las propias de un curso de primer nivel de programación. A lo largo de la asignatura se van introduciendo de forma progresiva las estructuras y



herramientas necesarias en cada punto del temario y que están disponibles en cualquier lenguaje de programación de propósito general. Los conceptos más importantes que se adquieren son los siguientes:

- Conceptos básicos de programación e ingeniería de software
- Elementos básicos: valores, tipos, constantes y variables.
- Estructuras básicas: secuencia, selección e iteración.
- Programación estructurada.
- Funciones y procedimientos.
- Esquemas de recorrido y búsqueda.
- Definición de tipos de datos. Formaciones. Registros.
- Punteros y estructuras no acotadas.
- Tipos abstractos de datos (TAD).
- Módulos

Como vehículo para la enseñanza de la programación se utiliza el lenguaje C± (léase C-más-menos), que está constituido por un subconjunto de los lenguajes C y C++. Por lo tanto, cualquier programa escrito en el lenguaje C± se puede editar, compilar y ejecutar en un entorno de desarrollo para C/C++. Es muy importante usar un lenguaje real para poder programar de manera natural e inmediata el computador. Esto permite al alumno comprobar en la práctica que los ejemplos propuestos funcionan y así también adquirir como resultado del aprendizaje el conocimiento de un lenguaje de uso habitual en el desarrollo de software.

Los resultados del aprendizaje dentro de la materia de la asignatura son los siguientes:

Especificar problemas sencillos

Diseñar programas que resuelvan problemas bien especificados

Seleccionar y realizar la codificación más adecuada para un algoritmo

Realizar pruebas de validación y depuración de un programa dado

Dominar un lenguaje de programación

5.CONTENIDOS DE LA ASIGNATURA

Los contenidos de este curso se organizan en los tres módulos o unidades didácticas siguientes.

UNIDAD DIDÁCTICA I

TEMA 1. Introducción

- Máquinas y programas
- Programación e ingeniería de software
- Lenguajes de programación
- Compiladores e Intérpretes
- Modelos abstractos de cómputo
- Elementos de la programación imperativa
- Evolución de la programación

TEMA 2. Elementos básicos de programación

- Lenguaje C±
- Notación BNF
- Valores y tipos
- Representación de valores constantes
- Tipos predefinidos
- Expresiones aritméticas
- Operaciones de escritura simples
- Estructura de un programa completo



TEMA 3. Constantes y Variables

- Identificadores
- El vocabulario de C±
- Constantes
- Variables
- Sentencia de asignación
- Operaciones de lectura simple
- Estructura de un programa con constantes y variables

TEMA 4. Metodología de Desarrollo de Programas (I)

- La programación como resolución de problemas
- Descomposición de un problema en subproblemas
- Desarrollo de programas por refinamientos sucesivos
- Aspectos de estilo

TEMA 5. Estructuras Básicas de la Programación Imperativa

- Programación estructurada
- Expresiones condicionales
- Estructuras básicas en C±

TEMA 6. Metodología de Desarrollo de Programas (II)

- Desarrollo con esquemas de selección e iteración
- Ejemplos de desarrollo con esquemas
- Verificación de programas
- Eficiencia de programas. Complejidad algorítmica

UNIDAD DIDÁCTICA II

TEMA 7. Funciones y Procedimientos

- Concepto de subprograma
- Funciones
- Procedimientos
- Paso de argumentos
- Visibilidad. Estructura de bloques
- Recursividad de subprogramas
- Problemas en el uso de subprogramas

TEMA 8. Metodología de Desarrollo de Programas (III)

- Operaciones abstractas
- Desarrollo por refinamiento usando abstracciones
- Programas robustos

TEMA 9. Definición de tipos

- Tipos definidos
- Tipo enumerado
- El tipo predefinido bool
- Tipos estructurados
- Tipo formación y su necesidad
- Tipo vector
- Vector de caracteres: Cadena (*String*)
- Tipo tupla y su necesidad
- Tipo registro (*struct*)

TEMA 10. Ampliación de estructuras de control

- Estructuras complementarias de iteración
- Estructuras complementarias de selección
- Equivalencia entre estructuras

TEMA 11. Estructuras de datos



Argumentos de tipo vector abierto
Formaciones anidadas. Matrices
El esquema unión
Esquemas de datos y esquemas de acciones
Estructuras combinadas

UNIDAD DIDÁCTICA III

TEMA 12. Esquemas típicos de operación con formaciones

Esquema de recorrido
Búsqueda secuencial
Inserción
Ordenación por inserción directa
Búsqueda por dicotomía
Simplificación de las condiciones de contorno

TEMA 13. Punteros y variables dinámicas

Estructuras de datos no acotadas
La estructura secuencia
Variables dinámicas
Realización de secuencias mediante punteros
Punteros y paso de argumentos
Punteros y vectores en C y C++

TEMA 14. Tipos abstractos de datos

Concepto de tipo abstracto de datos (TAD)
Realización de tipos abstractos en C++
Metodología de desarrollo basada en abstracciones

TEMA 15. Módulos

Concepto de módulo
Módulos en C++
Desarrollo modular basado en abstracciones

6.EQUIPO DOCENTE

- [ISMAEL ABAD CARDIEL](#)
- [JOSE ANTONIO CERRADA SOMOLINOS](#)
- [RUBEN HERADIO GIL](#)

7.METODOLOGÍA Y ACTIVIDADES DE APRENDIZAJE

La metodología seguida para el aprendizaje de esta asignatura es la propia de una universidad a distancia, que se caracteriza por el empleo conjunto de medios impresos, audiovisuales y de las nuevas tecnologías. Los materiales docente específicos, las comunidades virtuales de aprendizaje, la asistencia presencial a los estudiantes a través de los profesores tutores de los Centros Asociados y el uso de los diversos sistemas de comunicación (teléfono, videoconferencia, radio, televisión, correo electrónico, etc.) son los medios con que cuenta la UNED para la enseñanza a distancia y todos ellos son utilizados en esta asignatura.

Adicionalmente, el equipo docente mantendrá actualizada una página web completamente específica para la asignatura en la dirección:

<http://www.issi.uned.es/fp/>



8.EVALUACIÓN

La evaluación de la asignatura se realizará mediante pruebas presenciales y una evaluación continua basada en la realización de unas prácticas con un grado de dificultad creciente.

Evaluación continua: Prácticas.

Las prácticas son voluntarias y su realización podrá incrementar la nota de la prueba presencial hasta un máximo de UN PUNTO, pero sólo cuando en el test se tenga 6 ó más respuestas correctas. Es necesaria la realización correcta de las CUATRO prácticas para optar al incremento del punto y sólo necesitar las 6 respuestas correctas. Las prácticas se podrán realizar en el aula informática del Centro Asociado o en cualquier computador al que pudiera tener acceso el alumno utilizando el entorno de programación para el lenguaje C±.

La entrega y evaluación de las prácticas sólo se podrá realizar antes de presentarse al examen de la convocatoria de febrero con la fecha tope que se publicará en la página web de la asignatura. Si no se entregan las prácticas antes de dicha fecha se entiende que se está renunciando a la evaluación continua del curso.

La instalación del entorno de programación, el manejo básico y avanzado del entorno de programación, los enunciados de las prácticas, el método de corrección de las prácticas, etc., están descritos en el libro indicado en la bibliografía básica: Prácticas de Programación en C±.

Pruebas Presenciales: Exámenes.

Constará de dos partes:

- Primera parte: 10 preguntas tipo test (3 puntos).
- Segunda parte: Ejercicio (6 puntos).

Las respuestas del test se marcarán en la hoja de lectura óptica que se entregará en el examen. El test es eliminatorio y se necesitan al menos 8 respuestas correctas (6 respuestas si se han realizado las prácticas).

La solución final al ejercicio de la segunda parte se deberá escribir en el espacio reservado a tal fin. Se podrá utilizar todo el papel que se considere necesario, pero sólo se deberá entregar la solución final en dicho espacio reservado.

Para realizar la prueba personal no se permitirá el uso de ningún material auxiliar. En la página web de la asignatura se mantendrá actualizada toda la información de interés para el alumno con los posibles cambios que se pudieran introducir a lo largo del curso. Además, en esta página se podrán consultar las soluciones de los exámenes, las notas de las prácticas y las notas de los exámenes.

9.BIBLIOGRAFÍA BÁSICA

Comentarios y anexos:

El libro de texto base por el que se sigue completamente la asignatura es:

Título: Fundamentos de Programación.

Autores: José A. CERRADA y Manuel. COLLADO

Edición: Madrid 2010

Editorial Centro de Estudios Ramón Areces, S.A. Madrid, Tel.: 91 506 11 90. Fax: 91 468 19 52.

Para la realización de las prácticas el libro de texto es el siguiente:

Título: Practicas de Programación en C±

Edición: Madrid 2010.

Autores: Ismael ABAD, José A. CERRADA, Rubén HERADIO y Manuel COLLADO

Edición: Madrid 2010

Editorial Centro de Estudios Ramón Areces, S.A. Madrid, Tel.: 91 506 11 90. Fax: 91 468 19 52.



10.BIBLIOGRAFÍA COMPLEMENTARIA

Comentarios y anexos:

Las siguientes obras pueden ser de alguna utilidad para ampliar la preparación de la asignatura. No obstante, el texto base se ajusta completamente al programa.

Miguel Ángel ACERA GARCÍA: C/C++. Edición revisada y actualizada 2010. Anaya Multimedia, 2009

José Luis BALCAZAR: Programación metódica. McGraw-Hill, 2001.

Francisco J. CEBALLOS SIERRA: C/C++ Curso de programación. Tercera edición. Editorial RA-MA, 2007.

Harvey M. DEITEL y Paul J. DEITEL: Como programar C++. Prentice Hall Mexico, 2003.

José Rafael GARCÍA-BERMEJO: Programación estructurada en C. Pearson Educacion. 2008

Luis JOYANES AGUILAR: Programación en C++. Algoritmos, estructuras de datos y objetos. Segunda edición. McGraw-Hill, 2006.

Brian W. KERNIGHAN y Dennis M. RITCHIE: El lenguaje de programación C. Segunda edición. Prentice-Hall Iberoamericana. 1991.

Ricardo PEÑA MARÍ: Diseño de programas. Formalismo y abstracción. Tercera edición. Pearson Educacion, 2005.

Bjarne STROUSTRUP: El lenguaje de programación C++. , 2002.

11.RECURSOS DE APOYO

Todos los recursos de apoyo adicionales para el estudio de la asignatura se pondrán a disposición de los alumnos en la página web:

<http://www.issi.uned.es/fp/>

en la que se mantendrá actualizada la última versión del software del entorno de programación necesario para la realización de los ejercicios y las prácticas.

12.TUTORIZACIÓN

La tutorización presencial de la asignatura se realizará en los Centros Asociados correspondientes. La tutorización virtual o seguimiento de los aprendizajes se realiza a través del Curso Virtual de la asignatura, implantado en la plataforma oficial de la UNED para enseñanzas oficiales. A dicha plataforma se accede a través de la página principal de la Web de la UNED, mediante las claves que se facilitan al formalizar la matrícula.

Para cualquier consulta adicional, el horario de atención al alumno es:

Los días de la semana indicados junto a los teléfonos en horario de 16 a 20 h.



Lugar: Locales de la ETSI Informática de la UNED.

Edificio Interfacultativo.

C/ Juan del Rosal, 16.

Departamento de Ingeniería de Software y Sistemas Informáticos.

Tels 91 398 86 54 (Lunes)

91 398 82 42 (Jueves de 10 a 14 h.)

91 398 64 78 (Jueves)

Ámbito: GUI - La autenticidad, validez e integridad de este documento puede ser verificada mediante el "Código Seguro de Verificación (CSV)" en la dirección <https://sede.uned.es/valida/>



6B6EA749CEDEC08038D83B167DFB6848