

30204 - Programación 1

Información del Plan Docente

Año académico	2017/18
Centro académico	110 - Escuela de Ingeniería y Arquitectura 326 - Escuela Universitaria Politécnica de Teruel
Titulación	439 - Graduado en Ingeniería Informática 443 - Graduado en Ingeniería Informática
Créditos	6.0
Curso	1
Periodo de impartición	Primer Semestre
Clase de asignatura	Formación básica
Módulo	---

1. Información Básica

1.1. Introducción

El principal objetivo de la asignatura es el diseño de programas de ordenador que resuelvan problemas de tratamiento de información. En la asignatura se presentan herramientas para representar información y para tratarla, junto con metodologías para facilitar y sistematizar el análisis de problemas y el diseño de programas que los resuelvan.

1.2. Recomendaciones para cursar la asignatura

Es conveniente que el alumno que curse esta asignatura tenga facilidad para la comprensión y el análisis de problemas y la deducción lógica de soluciones. Una adecuada formación matemática en los estudios previos resulta muy conveniente.

1.3. Contexto y sentido de la asignatura en la titulación

Programación 1 tiene carácter de asignatura de formación básica. Es el primer paso de un camino cuyo destino es el aprendizaje de la programación de computadores. Lo aprendido en esta asignatura tiene continuidad en las asignaturas de la materia común de Computación y Programación.

La programación es uno de los pilares básicos de la formación de un Ingeniero en Informática y su aprendizaje es imprescindible para estudiar casi cualquier otra materia de las que intervienen en los estudios.

1.4. Actividades y fechas clave de la asignatura

El calendario de exámenes será el que establezca la Escuela y las fechas de entrega de trabajos de evaluación se anunciarán con la suficiente antelación.

2. Resultados de aprendizaje

2.1. Resultados de aprendizaje que definen la asignatura

El estudiante, para superar esta asignatura, deberá demostrar los siguientes resultados:

30204 - Programación 1

1. Conoce conceptos básicos ligados a la programación y la informática.
2. Comprende, analiza y resuelve problemas de tratamiento de información de complejidad baja o media y construye algoritmos que los resuelven.
3. Define las estructuras de datos más adecuadas para representar la información asociada a cada problema.
4. Diseña de forma descendente y documenta las acciones algorítmicas que resuelven cada problema de forma eficaz y eficiente.
5. Conoce algoritmos para resolver los problemas más frecuentes que se presentan al trabajar con estructuras de datos secuenciales e indexadas.
6. Conoce y comprende la sintaxis y la semántica de las construcciones básicas de un lenguaje de programación.
7. Escribe programas con buen estilo, con una documentación adecuada, con los comentarios precisos y con las especificaciones necesarias.
8. Sabe utilizar herramientas de edición, compilación, depuración y ejecución para desarrollar programas, así como sistemas operativos, bases de datos y otros programas con aplicación en ingeniería.
9. Utiliza estrategias para corregir los programas cuando no funcionan bien.
10. Pone a punto y ejecuta en un computador programas escritos en un lenguaje de programación determinado.

2.2.Importancia de los resultados de aprendizaje

El conjunto de los resultados de aprendizaje se pueden resumir diciendo que el alumno será capaz de analizar y resolver problemas de tratamiento de información de pequeña o media complejidad y construir los correspondientes programas en el lenguaje utilizado en la asignatura. Haber aprendido a hacerlo bien y con eficiencia es de vital importancia en el contexto de unos estudios de Ingeniería informática de los que uno de sus pilares es la Programación de computadores.

3.Objetivos y competencias

3.1.Objetivos

La asignatura y sus resultados previstos responden a los siguientes planteamientos y objetivos:

Planteamientos

- Se presentan los conocimientos básicos ligados a la programación y la informática
- Se presenta una metodología de análisis y resolución de problemas de tratamiento de información.
- Se presenta, a un nivel básico, la tecnología que la programación estructurada ofrece al programador.
- Se enseña, desde un punto de vista práctico, al alumno a completar el proceso que se inicia con la comprensión del problema, continúa con su análisis y resolución, prosigue con el desarrollo y puesta a punto del programa que permite resolverlo y concluye con la explotación y mantenimiento del programa.

Objetivos

- Que el alumno conozca los conceptos básicos ligados a la programación.
- Que el alumno sepa analizar problemas concretos, resolverlos y desarrollar programas que permitan obtener los resultados correspondientes en un computador.
- Que el alumno aprenda a utilizar y se familiarice con un entorno tecnológico para el desarrollo de programas.
- Que el alumno se ejercite como programador de computadores de forma individual y en equipo.

3.2.Competencias

Al superar la asignatura, el estudiante será más competente para:

1. Resolver problemas y tomar decisiones con iniciativa, creatividad y razonamiento crítico.
2. Comunicar y transmitir conocimientos, habilidades y destrezas en castellano.
3. Usar las técnicas, habilidades y herramientas de la Ingeniería necesarias para la práctica de la misma.
4. Aprender de forma continuada y desarrollar estrategias de aprendizaje autónomo.
5. Aplicar las tecnologías de la información y las comunicaciones en la Ingeniería.
6. Usar ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

4.Evaluación

30204 - Programación 1

4.1. Tipo de pruebas, criterios de evaluación y niveles de exigencia

El estudiante deberá demostrar que ha alcanzado los resultados de aprendizaje previstos mediante las siguientes actividades de evaluación

En la Escuela de Ingeniería y Arquitectura del Campus Río Ebro:

La evaluación de la asignatura en la convocatoria de febrero consta de tres pruebas:

- **Examen escrito** en el que el alumno ha de responder, en su caso, preguntas conceptuales y resolver problemas. Pondera un 70% en la nota de la asignatura. Es necesario obtener en este examen escrito una nota mínima de 4.0 puntos para optar a aprobar la asignatura. En caso de no alcanzar dicha puntuación mínima, la calificación de este examen escrito es la que constará en el acta de la asignatura.
- **Examen práctico** en el que el alumno ha de realizar individualmente un pequeño trabajo de programación en el laboratorio. Pondera un 15% en la nota de la asignatura.
- **Trabajo de programación** que será propuesto por los profesores de la asignatura con un plazo de entrega determinado. Los profesores indicarán si este trabajo debe realizarse individualmente o en equipo. Pondera un 15% en la nota de la asignatura.

Se considerarán no presentados en esta convocatoria, exclusivamente los alumnos que no se presenten al examen escrito.

Opcionalmente, a lo largo del cuatrimestre, los profesores de la asignatura podrán proponer una o más pruebas voluntarias de evaluación, a resolver por escrito en el aula o fuera de ella. Si su calificación total es igual o superior a 5 sobre 10, un 10% de la misma, hasta un máximo de 1 punto, se sumará a la calificación obtenida en la convocatoria de febrero en el caso de haber aprobado la asignatura. Esta mejora de la calificación no se aplica a quienes no hayan aprobado la asignatura en febrero o no hayan presentado todas las pruebas voluntarias y no se conserva para la convocatoria de septiembre.

La evaluación de la asignatura en la convocatoria de septiembre consta de dos pruebas:

- **Examen escrito** en el que el alumno ha de responder, en su caso, preguntas conceptuales y resolver problemas. Pondera un 70% en la nota de la asignatura. Es necesario obtener en este examen escrito una nota mínima de 4.0 puntos para optar a aprobar la asignatura. En caso de no alcanzar dicha puntuación mínima, la calificación de este examen escrito es la que constará en el acta de la asignatura.
- **Examen práctico** en el que el alumno ha de realizar individualmente un pequeño trabajo de programación en el laboratorio. Pondera un 30% en la nota de la asignatura.

Calificaciones obtenidas en febrero válidas en septiembre:

- Al alumno que opte por no realizar el examen práctico de septiembre, se le aplicará como calificación la suma de las calificaciones obtenidas en el examen práctico de febrero y en el trabajo de programación de febrero.
- Al alumno que opte por realizar el examen práctico de septiembre y que opte por no realizar el examen escrito de septiembre, se le aplicará como calificación del examen escrito la obtenida en el examen escrito de febrero.

Se considerarán no presentados en esta convocatoria, los alumnos que no se presenten ni al examen escrito ni al examen práctico de septiembre.

En la Escuela Universitaria Politécnica del Campus de Teruel:

Convocatoria de Febrero . La evaluación de la asignatura se realiza en base a dos pruebas:

P1. Prueba en la que el alumno deberá responder cuestiones y resolver ejercicios y problemas, se podrá requerir que el alumno resuelva dichos problemas trabajando en el computador o de forma escrita. Se requiere una nota mínima de 4.0 puntos en esta prueba para aprobar la asignatura. Si se obtiene esta nota mínima, entonces la prueba pondera un 70% en la nota de la asignatura y, si no se alcanza este mínimo, entonces la calificación en la asignatura es la de esta

30204 - Programación 1

prueba.

P2. Trabajos y prueba de programación con computador. Esta prueba pondera un 30% en la nota de la asignatura. Cada alumno deberá entregar los trabajos de programación que se indiquen en las prácticas de la asignatura. A la hora de calificar estos trabajos se valorará su funcionamiento según especificaciones, la calidad de su diseño y su presentación, la adecuada aplicación de los métodos de resolución, el tiempo empleado, así como la capacidad de los alumnos para explicar y justificar el diseño realizado.

Pruebas con carácter voluntario. A lo largo del cuatrimestre se plantearán varias pruebas voluntarias consistentes en la resolución, por escrito en clase o en casa, de ejercicios y problemas de programación. El 10% de la calificación de estas pruebas, es decir, entre 0 y 1.0 puntos, se sumará a la calificación obtenida por el alumno en la convocatoria de Febrero de la asignatura (no en la de Septiembre), siempre y cuando la nota de la asignatura sea superior a 4.5 ($0.7 \cdot P1 + 0.3 \cdot P2 \geq 4.5$).

Convocatoria de Septiembre. La evaluación de la asignatura se realiza en base a dos pruebas análogas a las de la convocatoria de Febrero, con las mismas ponderaciones y exigencia de notas mínimas. Las calificaciones del alumno obtenidas en la convocatoria de Febrero en cualquier de las pruebas (P1 y P2) se mantienen en Septiembre, salvo que el alumno opte por presentarse a la prueba correspondiente en esta nueva convocatoria, en cuyo caso prevalecerá la nueva calificación.

5. Metodología, actividades, programa y recursos

5.1. Presentación metodológica general

El proceso de aprendizaje que se ha diseñado para esta asignatura se basa en lo siguiente:

1. El estudio y trabajo continuado desde el primer día de clase.
2. El aprendizaje de conceptos y metodologías de diseño de programas a través de las clases magistrales impartidas en el aula, en las que se favorecerá la participación de los alumnos.
3. La aplicación de tales conocimientos al diseño de algoritmos y programas en las clases de problemas. En estas clases los alumnos desempeñarán un papel activo en la discusión y resolución de los problemas. En algunas de estas clases el trabajo de cada alumno podría ser evaluado, tal como se ha señalado anteriormente.
4. Las clases de prácticas en laboratorio en las que el alumno aprenderá la tecnología necesaria para poner a punto programas utilizando un determinado lenguaje de programación. Aprenderá asimismo a trabajar en un determinado sistema operativo con un conjunto de herramientas software para el desarrollo de programas.
5. Una parte de los trabajos de programación desarrollados en las prácticas de la asignatura se realizarán en equipo.
6. El aprendizaje de la programación exige un trabajo continuado por parte del alumno en el que se conjuga la comprensión de conceptos, el análisis y la resolución de problemas de programación utilizando "lápiz y papel" y la puesta a punto en computador de un buen número de programas.

5.2. Actividades de aprendizaje

El programa que se ofrece al estudiante para ayudarle a lograr los resultados previstos comprende las siguientes actividades...

En las clases impartidas en el aula se desarrollará el programa de la asignatura.

En las clases de problemas se resolverán problemas de aplicación de los conceptos y técnicas presentadas en el programa de la asignatura.

Las sesiones de prácticas se desarrollan en un laboratorio informático. En cada sesión el alumno deberá realizar un trabajo de programación, individualmente o en equipo, poniendo a punto uno o varios programas.

5.3. Programa

I. Conceptos y elementos básicos de Programación

- Problemas de tratamiento de información, algoritmos y programas
- Lenguajes de programación y ejecución de un programa
- Información, datos, operaciones y expresiones

II. Diseño de los primeros programas

- Diseño de algunos programas elementales
- Instrucciones simples y estructuradas
- Problemas de cálculo con enteros
- Desarrollo modular y descendente de programas
- Problemas de cálculo con números reales

III. Diseño de programas que trabajan con estructuras de datos

- Estructuración indexada de datos
- Representación de cadenas de caracteres
- Estructuración agregada de datos
- Algoritmos básicos de trabajo con estructuras de datos indexadas

IV. Diseño de programas que trabajan con datos almacenados en ficheros

- Entrada y salida de datos
- Trabajo con ficheros de texto
- Trabajo con ficheros binarios
- Trabajo con ficheros: otras posibilidades

V. Metodología de diseño de programas

5.4. Planificación y calendario

Calendario de sesiones presenciales y presentación de trabajos

La organización docente de la asignatura prevista en el campus Río Ebro es la siguiente.

- Clases teóricas (2 horas semanales)
- Clases de problemas (1 hora semanal)
- Clases prácticas de laboratorio (2 horas cada dos semanas). Son sesiones de trabajo de programación en laboratorio, tuteladas por un profesor, en las que participan los alumnos en grupos reducidos.

La organización docente de la asignatura prevista en el campus de Teruel es la siguiente.

- Clases teóricas (2 horas semanales)
- Clases de problemas y de prácticas de laboratorio (2 horas semanales)

Los horarios de todas las clases y las fechas de las sesiones de prácticas se anunciarán con suficiente antelación a través de las webs del centro y de la asignatura.

Los trabajos de programación propuestos en prácticas serán entregados al finalizar el cuatrimestre, en las fechas que se señalen.

5.5. Bibliografía y recursos recomendados

Teruel:

BB

Deitel, Paul J.. C++ : cómo programar /

30204 - Programación 1

Paul Deitel, Harvey Deitel ; traducción
Alfonso Vidal Romero Elizondo ; revisión
técnica Sergio Fuenlabrada Velázquez ...
[et al.]. - 9ª ed. Naucalpan de Juárez
(Estado de México) : Pearson, 2014

BB

Martínez Rodríguez, Javier. Javier
Martínez. Curso de Programación 1 /
Martínez Rodríguez, Javier. Servicio de
reprografía de la EINA. 2015

LISTADO DE URLs:

Documentación sobre el lenguaje C++
[<http://www.cplusplus.com/>]
Documentación sobre los lenguajes C y
C++
[<http://www.cprogramming.com/>]
Entorno de ejecución en línea «C++ shell»
[<http://cpp.sh/>]
Latre, Miguel Ángel. Problemas de
Programación 1 / Miguel Ángel Latre y
Javier Martínez. Servicio de reprografía de
la EINA y web de la asignatura. 2015
[<http://webdiis.unizar.es/asignaturas/PROG1/doc/materiales/problemasProg1.pdf>]
Manual en línea de bibliotecas
predefinidas en C++
[<http://www.cplusplus.com/reference/>]
Miguel Ángel Latre y Javier Martínez.
Prácticas de Programación 1. Servicio de
reprografía de la EINA y web de la
asignatura. 2015
[<http://webdiis.unizar.es/asignaturas/PROG1/doc/materiales/practicasProg1.pdf>]
Página web de la asignatura
[<http://webdiis.unizar.es/asignaturas/PROG1/>]
Wikilibro de Programación en C++
[http://es.wikibooks.org/wiki/Programaci%C3%B3n_en_C++]