

E3TOOL: HACIA EL ESPACIO EUROPEO DE ENSEÑANZA SUPERIOR. HERRAMIENTAS DE APOYO A LA PLANIFICACIÓN DOCENTE PARA LA IMPLANTACIÓN DE LAS DIRECTRICES DE BOLOGNA EN LA UNIVERSIDAD DE ALICANTE

M. Saiz Noeda, , P. J. Ponce de León, J. L. Verdú, J. L. Vicedo

*Departamento de Lenguajes y Sistemas Informáticos
Universidad de Alicante*

RESUMEN

Se presenta en este artículo el contexto, los objetivos y los resultados del trabajo llevado a cabo por los integrantes del proyecto *E3TOOL: Hacia el Espacio Europeo de Enseñanza Superior. Herramientas de apoyo a la planificación docente para la implantación de las directrices de Bologna en la Universidad de Alicante* ubicado en la convocatoria de Proyectos de Investigación en Docencia Universitaria del curso 2003-2004 propuestos por el Instituto de Ciencias de la Educación de la Universidad de Alicante.

En este trabajo se presenta en detalle la evolución de la red en sus contenidos y objetivos, el marco teórico en el que se encuadra, los resultados obtenidos y una reflexión sobre los mismos.

INTRODUCCIÓN

Este artículo recoge los objetivos y resultados del proyecto de investigación en docencia universitaria *E3TOOL: Hacia el Espacio Europeo de Enseñanza Superior: herramientas de apoyo a la planificación docente para la implantación de las directrices de Bologna en la Universidad de Alicante*. El proyecto se encuadra dentro del marco conceptual del *Espacio Europeo de Educación Superior (EEES)*. En este marco, la Declaración de Bologna de 1999 incluye entre sus principales objetivos el establecimiento de un Sistema Europeo de Transferencia de Créditos (ECTS), estableciendo como fecha límite el año 2010 para el desarrollo del EEES. Las reuniones posteriores no han hecho sino ratificar la declaración de Bologna en cada uno de sus puntos.

El crédito europeo de transferencia y acumulación (el llamado crédito ECTS) aparece, por tanto, como un punto de referencia para lograr la transparencia y calidad en la formación que se exigen en la actualidad.

El sistema ECTS se basa en algunos elementos básicos:

- la utilización de créditos ECTS como valores que representan el volumen de trabajo efectivo del estudiante y el rendimiento obtenido mediante calificaciones comparables (grados ECTS). El crédito ECTS se basa en el volumen total de trabajo del estudiante y no se limita exclusivamente a las horas de asistencia en clases presenciales.
- la información sobre los programas de estudios y los resultados de los estudiantes con documentos con un formato normalizado (guía docente y certificados académicos).

Se debe destacar que, en el marco del ECTS, 60 créditos representan el volumen de trabajo de un año académico. Un crédito europeo representa entre 25 y 30 horas de trabajo del estudiante, siendo el método recomendado para la asignación de créditos seguir un procedimiento descendente (*top-down*).

Tanto en el *Estudio sobre la asignación de créditos europeos a las distintas materias del currículum de matemáticas*, incluido en el *Documento de trabajo sobre la integración de los estudios españoles de matemáticas en el espacio europeo de enseñanza superior* (Euromates, 2003), como en el *Informe técnico sobre el crédito europeo y el sistema educativo español* (Pagani, 2002), se recomienda la utilización de la guía docente para la asignación de créditos ECTS, ya que ésta asigna factores a cada materia que relacionan el número de horas presenciales y el número de horas de trabajo personal del estudiante. Este factor, para asignaturas con un número de horas de prácticas similar al número de horas de teoría, como la que nos ocupa, es aproximadamente 1,5 horas de estudio por hora presencial (en teoría y práctica).

A continuación se presenta el método y proceso de investigación, donde se pone de manifiesto el trabajo realizado por los integrantes de la red, en el que se detallan los objetivos iniciales planteados por la red original así como la evolución que ésta ha sufrido a lo largo de su desarrollo. Así mismo, se muestran las interacciones que en forma de reuniones de trabajo ha tenido la red y las repercusiones que la asistencia a los diferentes cursos propuestos por el Instituto de Ciencias de la Educación (ICE en adelante) de la Universidad de Alicante han tenido en la propuesta final.

Se analizan además los resultados obtenidos de la investigación docente así como un conjunto de conclusiones relativas a dichos resultados.

La última parte del documento, recogida en un anexo, presenta propiamente la guía docente de la asignatura *Diseño y Análisis de Algoritmos*, estructurada según las recomendaciones de la UE: objetivos de la asignatura, competencias a adquirir, contenidos, bibliografía, planteamiento docente, evaluación del aprendizaje y plan de trabajo de los alumnos. Esta guía docente ha de servir como herramienta base para la implantación del sistema de créditos ECTS en la asignatura objeto del presente trabajo.

Es nuestro deseo y ha sido nuestro objetivo durante el desarrollo de la investigación, que el trabajo aquí realizado sea trasladable, al menos en parte, al resto de las asignaturas de las titulaciones de Informática.

1. MÉTODO Y PROCESO DE INVESTIGACIÓN

1.1 PARTICIPANTES

Los profesores que participan en esta red pertenecen al Departamento de Lenguajes y Sistemas Informáticos. Todos ellos tienen amplia experiencia docente en asignaturas de programación vinculadas a la que es objeto de esta experiencia siendo además docentes de la asignatura específica objeto de estudio.

1.2 ORÍGENES Y EVOLUCIÓN DE LA RED

La red docente E3TOOL ha sufrido a lo largo del periodo de su desarrollo, y con la madurez de sus participantes, algunos cambios considerables desde su concepción original.

La red, tal y como fue diseñada en sus orígenes, suponía un punto de partida en el desarrollo de un conjunto de protocolos de actuación para determinar cuantitativa y cualitativamente los parámetros involucrados en la adaptación de contenidos docentes al nuevo espacio europeo de enseñanza superior. Así, E3TOOL proponía el desarrollo de un conjunto de herramientas de apoyo a la investigación y planificación de la docencia para la elaboración curricular de contenidos adaptados al nuevo crédito europeo.

Se interpretaba por tanto como una "metared" en la que los contenidos desarrollados podían ser concebidos no sólo como un proyecto de pre-acción cuyo objetivo es el de "elaborar el mapa de competencias -conocimientos, capacidades y destrezas- que configuren una materia concreta", sino como un conjunto de elementos de ayuda para la elaboración de dichos mapas de competencia en cualquier materia. Estos resultados podrían servir de elementos de referencia y apoyo a cualquier otra red o grupo de trabajo que desee investigar, planificar, desarrollar y evaluar objetivos, competencias y contenidos para cualquier materia docente.

Sin embargo, y a partir de las reuniones de organización de la red y la asistencia a los diferentes cursos (que más adelante se mencionarán) propuestos por el ICE y destinados a la orientación de los investigadores en las diferentes áreas de trabajo propuestas, el (reducido) equipo integrante de la red determinó la imposibilidad de desarrollar en un sólo año un proyecto tan ambicioso. Por ello, y tras realizar las consultas oportunas a los responsables de la convocatoria, se tomó la decisión de producir de forma gradual los objetivos propuestos originalmente y convertir así este proyecto en una red trianual en la que se siguiera la metodología propuesta en las diferentes áreas de trabajo.

De esta manera, durante el primer año, y tomando como referencia una materia troncal y una asignatura de tercer curso de la Ingeniería Informática (Diseño y Análisis de Algoritmos), se desarrollaría su guía docente. A continuación, durante el segundo año, se establecerían los parámetros y se cumplirían los objetivos para determinar el tiempo y el esfuerzo que los alumnos dedican a la mencionada materia para así, durante el tercer año, poder desarrollar propiamente los protocolos y herramientas que se plantearon originalmente en la red. Este trabajo, enfocado de manera gradual, no sólo permitiría abarcar el problema de manera más acorde a las características del grupo humano integrante de la red, sino que haría posible abordar los objetivos desde una perspectiva ostensiblemente más madura desde el conocimiento profundo del espíritu del espacio europeo de enseñanza superior.

A continuación se presentarán los contenidos y resultados básicos de las diferentes reuniones que el equipo de trabajo de la red ha llevado a cabo. Así mismo, se realizará un breve resumen de lo más destacable en cuanto a la asistencia a los diferentes cursos que el Instituto de Ciencias de la Educación ha organizado para la orientación de los investigadores vinculados a las redes docentes del presente año.

1.3 PLAN DE TRABAJO

El plan de trabajo ha sido desarrollado a lo largo del curso 2003-2004 y parte del 2004-2005. La espina dorsal de este plan de trabajo ha sido el conjunto de reuniones convocadas entre los miembros de la red en las que se ha puesto de manifiesto las excelentes condiciones de los integrantes del grupo en su trabajo en equipo. A continuación se muestra un resumen de las reuniones más importantes llevadas a cabo por parte de los integrantes de la red E3TOOL con su distribución cronológica:

- Los integrantes de la red, que ya participaron en convocatorias anteriores, se reúnen para consensuar la posibilidad de concurrir en la convocatoria actual cuyo objetivo: "Participar en la conformación del Espacio Europeo de Educación Superior" les atrae por su relevancia para el futuro de la enseñanza universitaria.
- Decisión del tema objeto del estudio, elección de la asignatura, nombre de la red docente y elaboración de un borrador para cumplimentar la solicitud de participación en el programa de formación e investigación docente. Se

plantea la necesidad inicial de desarrollar un conjunto de herramientas de apoyo a la investigación y planificación de la docencia. El objetivo de la red se encamina en este sentido.

- Complimentación de dicha solicitud y envío al ICE.
- Planificación y reparto del trabajo. Dadas las especiales características de la red propuesta, y debido a que no puede ser en principio encuadrada en ninguna de las áreas de trabajo específicas propuestas por el ICE, los participantes deciden acudir a los cursos propuestos para la orientación de los investigadores. Dos miembros de la red asistirán al curso de "Elaboración de Guías Docentes ECTS" mientras que los otros dos lo harán al de "Valoración del trabajo y esfuerzo de aprendizaje". De esta manera, en reuniones sucesivas, se podrán poner en común los planteamientos de ambos cursos y decidir sobre el conjunto de herramientas y objetivos a desarrollar en la red.
- Tras la asistencia a las primeras sesiones de los mencionados cursos, los miembros de la red coinciden en lo ambicioso del proyecto original y consideran la posibilidad de reorientar el trabajo de la red para poder abarcar mejor los objetivos propuestos. De esta forma se plantean unas líneas de actuación en tres fases: en una primera se elaborará el curriculum de la asignatura siguiendo las orientaciones del curso de elaboración de guías docentes; durante la segunda fase se trabajará sobre la valoración del tiempo y el esfuerzo de los alumnos para, en una tercera fase, en la que el trabajo habrá alcanzado una mayor definición y madurez, se tratará el desarrollo de herramientas según el planteamiento original. Se decide, por tanto, poner en común esta idea con los responsables del ICE.
- Tras haber consensuado con el ICE la nueva estrategia, y a partir de la asistencia a las diferentes sesiones de los cursos, los miembros de la red se reúnen para realizar una puesta en común del trabajo hecho y del pendiente. Cada miembro propone nuevas ideas de incorporación a la memoria docente y se elabora progresivamente un borrador con el compendio de todas ellas.
- Partiendo del borrador ya mencionado, se elabora un documento "casi" definitivo que se distribuye entre los miembros de la red para su revisión.
- A partir de las revisiones de todos los miembros, se prepara el documento definitivo para ser remitido al ICE.
- Tras una reunión con el ICE en la que se pone de manifiesto algunas carencias en el documento presentado, los miembros de la red deciden ampliar la memoria con algunas secciones que detallen aspectos relativos al trabajo realizado, a los objetivos conseguidos y reflexiones sobre los resultados y sus conclusiones.
- Se prepara y termina la memoria ampliada con los aspectos mencionados anteriormente para su envío definitivo al ICE.

1.4 ASISTENCIA A CURSOS DEL ICE

Tal y como se ha comentado, los miembros de la red hicieron un esfuerzo por asistir puntualmente a las reuniones de trabajo propuestas por el ICE en forma de cursos. Se plantearon dos líneas de trabajo correspondiendo a dos cursos diferentes: "Elaboración de Guías Docentes ECTS" y "Valoración del trabajo y esfuerzo de aprendizaje". Estos cursos, que comenzaron la primera semana de diciembre, se prolongaron hasta la segunda quincena de febrero de 2004.

La asistencia a los mencionados cursos posibilitó entrar en contacto con la realidad planteada por los promotores de la convocatoria de redes de este año y poder definir de manera más consensuada los objetivos y resultados esperados en cada una de las áreas de trabajo. Así mismo, también sirvieron para madurar la idea de la red original y, tal y como se ha detallado anteriormente, replantearla y redirigirla hacia objetivos más abordables.

Es importante destacar el esfuerzo que, para una red tan pequeña como esta (integrada por tan sólo cuatro investigadores) ha supuesto el planteamiento y la asistencia tan diversificada a las actividades propuestas por el ICE.

2. RESULTADOS

Una guía docente contiene información sobre los objetivos, contenidos y prerrequisitos de la asignatura, las competencias o destrezas a adquirir, la metodología de la enseñanza, el sistema de evaluación del aprendizaje y una estimación del tiempo de trabajo que un estudiante medio debe invertir para superar la asignatura. Se ha de tener en cuenta que, en el EEES, se entiende que no hay enseñanza si no hay aprendizaje, hecho que se reflejará en la guía planteando una metodología que fomente el aprendizaje mediante el trabajo personal o en grupo del alumno, contabilizando explícitamente estas horas de trabajo como parte del esfuerzo conducente a superar la asignatura.

La guía docente permitirá, además, la puesta en marcha de mecanismos de seguimiento de su puesta en práctica, mediante encuestas a alumnos y profesores u otros mecanismos de evaluación. Esto permitirá realizar las correcciones necesarias en los contenidos, metodología o sistema de evaluación de la asignatura, que conduzcan a que la guía refleje con la mayor precisión posible el volumen de trabajo real que cada asignatura requiere.

Los objetivos que se han planteado en este proyecto son la preparación de una guía docente para la asignatura *Diseño y Análisis de Algoritmos* que permita la asignación de créditos ECTS al conjunto de asignaturas del curso en el que se ubica la asignatura, una vez verificado su fiel reflejo en el desarrollo de la asignatura en próximos cursos, y establecida la relación definitiva entre horas de trabajo y créditos ECTS.

Para esta etapa de verificación, se prevé el desarrollo, en futuros proyectos de investigación en docencia universitaria, de herramientas que permitan automatizar el cálculo y asignación de créditos ECTS, a nivel de asignatura, curso y plan de estudios.

La guía docente, resultado de este proyecto, se ajusta al modelo establecido por la normativa de la UE e incluye los siguientes puntos:

- Identificación y contextualización de la asignatura
- Objetivos, competencias y prerrequisitos
- Contenidos y bibliografía
- Consideraciones metodológicas
- Plan de trabajo de los alumnos

Con ella se pretende no sólo establecer un modelo para la asignación de créditos ECTS a una asignatura que se ubica dentro del ámbito de la enseñanza de la metodología de la programación, sino enfatizar un planteamiento docente centrado en el aprendizaje, el cual entendemos es el futuro de la enseñanza superior en Europa. No se ha descuidado para ello el estudio de currículos internacionales que recogen la materia troncal objeto de estudio (ACM, 1968, 1979, 2001; UNESCO, 1984).

2.1 DISCUSIÓN DE LOS RESULTADOS

Se ha planteado una metodología docente, basada en el lema "no hay enseñanza sin aprendizaje", donde el esfuerzo del alumno está repartido al 50% entre contenido teórico y trabajo práctico, tanto en número de horas presenciales como no presenciales. Entendemos que en la asignatura *Diseño y análisis de algoritmos* la teoría y la práctica están íntimamente ligadas. De hecho, la mayoría de los contenidos teóricos son, en realidad, nociones y métodos de programación que el alumno ha de aplicar directamente a los problemas prácticos que se le plantean durante el curso.

Los tres bloques en que se divide el contenido de la asignatura se presentan de forma que los dos primeros introducen las nociones y herramientas que permitirán verificar la validez e idoneidad de los esquemas de programación presentados en el último y más extenso bloque.

El cálculo de complejidades introducido en el primer bloque servirá para poder distinguir qué esquemas funcionan mejor que otros y bajo qué condiciones lo hacen. Este bloque tiene asociado un trabajo práctico que servirá además para introducir las herramientas de programación que se utilizarán a lo largo de la asignatura.

La corrección de los algoritmos, presentada en el segundo bloque, introduce los métodos de verificación de un algoritmo que serán utilizados para comprobar la corrección de los esquemas presentados en el siguiente bloque. El tercer bloque presenta los esquemas de programación clásicos utilizados para diseñar algoritmos capaces de resolver problemas de gran complejidad mediante un diseño cuidadoso y sistemático de la solución. Corresponden a este bloque dos trabajos prácticos en los que se emplearán cada uno de los esquemas presentados.

El número de horas estimado para la presentación de cada bloque de contenidos está en relación directa al volumen de información que contienen y a su importancia relativa en el conjunto de la asignatura. Así, el último bloque es el más extenso, pues presenta la materia clave de la asignatura.

El contenido presentado en la guía docente *Diseño y análisis de algoritmos* cubre un espacio indispensable dentro de las titulaciones en las que se imparte la asignatura, así como en el cuadro de competencias profesionales que el egresado de estas titulaciones debe poseer. Algunas de esas competencias son:

- Conocer los esquemas clásicos de diseño de algoritmos para la búsqueda de soluciones.
- Saber identificar los problemas reales como susceptibles de ser resueltos mediante esos esquemas.
- Utilizar una filosofía de programación basada *diseñar antes de implementar*.
- Saber utilizar los esquemas tipo para la implementación de soluciones algorítmicas particulares.
- En general, adquirir buenos hábitos de programación.

Se ha tenido especial cuidado en no solapar contenidos con otras asignaturas afines, como las que figuran entre los prerrequisitos de *Diseño y análisis de algoritmos* o aquellas de las que ésta es prerrequisito, contribuyendo a un planteamiento docente de las titulaciones sistemático y bien estructurado.

Como ya se ha dicho, la metodología docente empleada se centra en el aprendizaje. Los alumnos cuentan con distintas herramientas de apoyo para su aprendizaje personal,

algunas de ellas derivadas del uso del *Campus Virtual*¹ (CV) de la Universidad de Alicante como medio de comunicación entre docentes y alumnado. Las tutorías vía CV son la herramienta clave que permite al alumno plantear las dudas desde su espacio de trabajo habitual o desde cualquier laboratorio de prácticas de la universidad. El tiempo medio de respuesta no suele ser superior a un día y en muchos casos es sólo de algunos minutos. El CV permite además un acceso inmediato y ágil a los materiales de la asignatura (apuntes de clase, enunciados de problemas, bibliografía,...). A todo este conjunto de herramientas disponibles en el CV hay que añadir el hecho de que las sesiones de prácticas en laboratorio son abiertas, permitiendo el acceso a un ordenador y a la realización de consultas al profesor en cualquier sesión, limitado, claro está, a la disponibilidad de puestos libres en el laboratorio.

Se podría decir que los contenidos de la asignatura constituyen una parte de la *materia clásica* de la metodología de la programación. Sin embargo, dado que los ejemplos de implementación se expresan mediante pseudocódigo, resulta sencillo adaptar los detalles de implementación a nuevos lenguajes de programación, al separar deliberadamente los detalles de diseño de una solución algorítmica de las peculiaridades de un lenguaje de programación determinado. Así por ejemplo, en los cursos hasta ahora impartidos, los alumnos pueden desarrollar los trabajos prácticos en lenguaje C o C++, a su elección, siendo los conocimientos adquiridos igualmente aplicables mediante otros lenguajes como Java, Perl, Pascal o C#.

Queda patente, a nuestro juicio, que esta asignatura integra en sus objetivos y metodología un uso intensivo de las TIC, tanto a nivel de la docencia presencial como no presencial: tutorías, bibliografía y materiales en CV, prácticas con ordenadores, apuntes y enunciados de problemas en formato electrónico, etc., factor éste clave en el Espacio Europeo de Educación Superior y las nuevas tendencias de docencia universitaria.

3. CONCLUSIONES

Se ha presentado, en esta memoria, la guía docente de la asignatura *Diseño y análisis de algoritmos*, impartida en las titulaciones de Informática de la Universidad de Alicante. La guía es una herramienta base para la puesta en marcha de mecanismos de seguimiento que permitan realizar una comparación realista de la valoración del esfuerzo estimado y el esfuerzo real. Basándonos en nuestra experiencia actual, se prevén pocos cambios en el programa de la asignatura, al tratarse de una materia clásica de la metodología de la programación, que no por clásica deja de estar vigente hoy día. De hecho, todos los esquemas presentados pueden obtener una solución óptima a un problema dado cuando se dan las condiciones de aplicación adecuadas. Algunos de estos esquemas son además la base de otros más complejos que se estudian en asignaturas de cursos superiores como *Algoritmia Avanzada*.

En cuanto a la metodología docente, cabe esperar un incremento en el uso de las TIC, con la puesta en marcha, por ejemplo, de foros virtuales donde alumnos y profesores discuten sobre cuestiones relacionadas con la teoría o prácticas de la asignatura. Resulta menos sencillo utilizar las TIC de forma íntegra en el proceso de evaluación, por los problemas derivados del uso de tecnología digital como la copia de trabajos a coste cero, la dificultad que entraña la identificación del alumno en pruebas de evaluación on-line, etc. Se impone, por estas razones, la necesidad de un examen

¹ Visitar <http://www.ua.es/es/univirtual/index.html> para ampliar la información sobre esta herramienta.

presencial teórico sobre la materia del curso que permita evaluar objetivamente los conocimientos del alumno.

La automatización de la corrección de los trabajos prácticos, sujeta en menor medida a los problemas de copia digital antes mencionados, es una mejora del sistema de evaluación que ya se ha comenzado a aplicar de forma parcial. Esta corrección semi-automática de las prácticas permite la obtención de resultados de evaluación en un plazo muy breve, que son luego discutidos con el alumno en sesiones de revisión de prácticas.

Esperamos que el objetivo de la guía docente de la asignatura *Diseño y análisis de algoritmos*, como vehículo para la correcta asignación de créditos ECTS al conjunto de asignaturas de tercer curso de las titulaciones de informática de la UA, se vea confirmado en una posterior etapa de verificación del esquema propuesto. Se pretende para ello desarrollar herramientas de apoyo a la asignación de créditos ECTS y a su verificación mediante los datos recogidos durante la evolución de cursos venideros.

4. BIBLIOGRAFÍA

ACM (1968) ACM curriculum committee on computer science. curriculum 68. Recommendations for the undergraduate program in Computer Science, *Communications of the ACM*, 11(3).

ACM (1979) ACM curriculum committee on computer science. curriculum 78. Recommendations for the undergraduate program in Computer Science, *Communications of the ACM*, 22(3).

ACM (2001) ACM code of ethics and professional conduct, *Association for Computing Machinery*.

Euromates (2003) Documento de trabajo sobre la integración de los estudios españoles de matemáticas en el espacio europeo de enseñanza superior, http://www.matematicas.us.es/anuncios/bolonia/documento_final.pdf.

Pagani, R. (2002) Informe técnico: el crédito europeo y el sistema educativo español, <http://www.eees.ua.es/ects/credito%20europeo.pdf>.

UNESCO (1984) A modular curriculum in computer science, UNESCO-IFIP.

ANEXO: GUÍA DOCENTE

En el presente anexo se desarrolla la guía docente de la asignatura objeto del estudio, *Diseño y Analisis de Algoritmos*. En esta guía docente, tras una contextualización de la asignatura se delimitarán los objetivos, competencias y prerrequisitos de la materia, así como sus contenidos teórico-prácticos. A continuación se describirán un conjunto de estrategias dirigidas al proceso de aprendizaje, finalizando con el plan de trabajo de los alumnos.

1. CONTEXTUALIZACIÓN

En este primer capítulo se define el ámbito del conocimiento en el que se enmarca la asignatura *Diseño y Análisis de Algoritmos* (DAA), objeto de esta memoria: la metodología de la programación (MP) y los diferentes aspectos de éste de cuyo estudio se encarga la asignatura, como introducción al perfil de ésta y su ubicación en el plan de estudios. La ficha de la asignatura tratada es la siguiente:

DISEÑO Y ANÁLISIS DE ALGORITMOS

Titulación: Ingeniería en Informática

Código: 9173

Tipo de asignatura: Troncal

Nivel: Primer ciclo

Curso: 3º

Carácter: Anual

Número de créditos: 6 (3 teóricos + 3 prácticos)

Se presentarán los contenidos generales de la asignatura troncal detallando además sus relaciones de incompatibilidad con otras asignaturas y sus vínculos con asignaturas recomendadas o que las recomiendan, con objeto de establecer la ubicación de la asignatura en los planes de estudio de las diferentes ingenierías en informática de la Universidad de Alicante.

1.1 LA DISCIPLINA DE METODOLOGÍA Y TECNOLOGÍA DE LA PROGRAMACIÓN

En la corta vida de la programación como disciplina científica, siempre ha estado presente la discusión acerca de cómo enfocar la resolución de problemas, es decir, cómo diseñar algoritmos que los resuelvan. La realidad es que podemos encontrar ejemplos de algoritmos que difícilmente se conocerían a no ser por la imaginación aplicada por sus autores. De hecho, algunos trabajos sobre el tema (Knuth, 1973) inciden sobre la importancia de la componente artesanal en la tarea de la obtención de algoritmos.

El estudio del proceso de resolución de problemas en programación, así como de las características y propiedades de las herramientas a utilizar, se conoce actualmente como *metodología de la programación*. Este término engloba desde cuestiones muy teóricas (abstracción de métodos o ideas) hasta las particularidades de los lenguajes de programación.

Entre los diferentes aspectos que merecen una atención especial en el ámbito del estudio de los algoritmos y de cuyo estudio se encarga la asignatura objeto de la presente memoria, destacamos los siguientes:

Especificación. Constituye el primer paso en la resolución metódica de problemas. Consiste en la obtención de un enunciado formal que sintetice los datos de los que se parte y aquellos resultados que se pretenden conseguir con la aplicación del algoritmo.

Diseño. Consiste en el estudio formal del problema y la aplicación de técnicas y/o esquemas que faciliten la obtención de un algoritmo que resuelva dicho problema.

Complejidad. Hace referencia a la estimación de la cantidad de recursos (tiempo y memoria) que el algoritmo necesitará para su ejecución una vez implementado. Este aspecto tiene una gran importancia tanto en la teoría como en la práctica. Además, para que este análisis tenga importancia, ha de ser independiente del computador y del lenguaje de implementación final que se vaya a utilizar.

Verificación. Engloba el estudio de las técnicas de comprobación formal de la corrección del algoritmo. Consiste en demostrar que el algoritmo termina en un tiempo finito y que su resultado es correcto para toda entrada de datos válida.

Codificación. Es el proceso de obtención de un programa correcto expresado en algún lenguaje de programación a partir del algoritmo. Este proceso puede llegar a ser bastante complejo en función del nivel de especificación del algoritmo y del lenguaje de programación utilizado.

1.2 ENSEÑANZA EN METODOLOGÍA Y TECNOLOGÍA DE LA PROGRAMACIÓN

Los Reales Decretos 1459, 1460 y 1461 de 26 de octubre de 1990 establecen respectivamente, los títulos oficiales de Ingeniero Técnico en Informática de Gestión, Ingeniero Técnico en Informática de Sistemas e Ingeniero en Informática así como las directrices generales propias de los planes de estudio conducentes a la obtención de cada uno de los títulos. Para cada titulación, dichas directrices definen las materias troncales y los créditos asignados a cada una de ellas.

Las directrices que definen la materia troncal objeto de este trabajo (Metodología y Tecnología de la Programación) son las siguientes:

- Diseño de Algoritmos
- Análisis de Algoritmos
- Lenguajes de Programación
- Diseño de programas: Descomposición modular y documentación
- Técnicas de verificación y pruebas de programas.

La materia troncal “Metodología y Tecnología de la Programación”, definida en las directrices propias de las tres titulaciones oficiales en Informática, se estructura en las siguientes asignaturas troncales definidas por la Universidad de Alicante:

- Fundamentos de Programación I
- Fundamentos de Programación II
- Diseño y Análisis de Algoritmos

En el siguiente cuadro aparecen, por titulación, los créditos totales de la materia troncal (según las directrices propias de la titulación) y los créditos por asignatura troncal asignados por la universidad. Entre paréntesis se indican los créditos adicionales que se incluyeron en el plan de estudios.

| <i>Titulación</i> | <i>Créditos Totales</i> | <i>Asignaturas</i> | <i>Créditos</i> |
|-------------------|-------------------------|---------------------------------|-----------------|
| ITIG | 15 | Fundamentos de Programación I | 6 (1) |
| | | Fundamentos de Programación II | 6 (1) |
| | | Diseño y Análisis de Algoritmos | 6 (1) |
| ITIS | 12 | Fundamentos de Programación I | 6 (2) |
| | | Fundamentos de Programación II | 6 (2) |
| | | Diseño y Análisis de Algoritmos | 6 (2) |
| II | 15 | Fundamentos de Programación I | 6 (1) |
| | | Fundamentos de Programación II | 6 (1) |
| | | Diseño y Análisis de Algoritmos | 6 (1) |

Los siguientes puntos tratarán los contenidos generales de estas asignaturas, así como las relaciones existentes entre ellas y el resto de las asignaturas de las titulaciones de Informática.

1.2.1 Fundamentos de Programación I y II (FPI y FPII)

Las asignaturas FPI y FPII constan cada una de 6 créditos (3 teóricos + 3 prácticos), son troncales y de carácter cuatrimestral (primero y segundo respectivamente) y se imparten en el primer curso de las tres titulaciones.

El objetivo principal de estas asignaturas consiste en facilitar al alumno una visión inicial del campo de la programación.

Se presenta el concepto de algoritmo así como sus distintas técnicas de representación y se efectúa una aproximación básica a los tipos generales de problemas susceptibles de resolución por ordenador y las herramientas necesarias para ello. Se introducen los elementos básicos de un lenguaje de programación de tercera generación: variables, funciones, tipos de datos estructurados, punteros, estructuras de datos dinámicas y ficheros. Se analizan las implicaciones del trabajo de programador, individualmente y como miembro de un equipo, pretendiendo que el alumno sea capaz de escribir en un plazo razonable, programas correctos, eficientes, estructurados, bien documentados y legibles. Se estudia el lenguaje de programación de alto nivel C y se introducen algunos aspectos básicos del lenguaje de programación orientado a objetos C++,

así como los conceptos esenciales de lenguaje algorítmico que verán su continuación en otras asignaturas de programación.

1.2.2 Diseño y Análisis de Algoritmos

Asignatura de 6 créditos (3 teóricos + 3 prácticos), troncal y de carácter cuatrimestral (2º cuatrimestre) que se imparte en el tercer curso de las tres titulaciones. Esta asignatura profundiza en el campo del análisis y diseño de algoritmos. En cuanto al análisis, se presentan los conceptos y técnicas básicas de verificación y prueba así como los relativos al cálculo de eficiencia de los algoritmos. En cuanto al diseño, se analizan diversas técnicas de diseño avanzadas: los algoritmos recursivos y la aplicación de esquemas algorítmicos a la resolución de problemas.

1.2.3 Relación con otras asignaturas

La materia troncal tratada establece un conjunto de relaciones de incompatibilidad¹ con otras asignaturas de las titulaciones de Informática. Además de estas relaciones de incompatibilidad, también establece vínculos con asignaturas que son recomendadas o que recomiendan a las integradas en ésta. El siguiente gráfico recoge el esquema general de todas las relaciones existentes con las asignaturas estudiadas².

¹ El concepto de incompatibilidad, materializado en el proceso de matriculación oficial, indica la existencia de prerequisites académicos imprescindibles para la comprensión de los contenidos de la asignatura incompatible

² Dado que las asignaturas tienen caracteres diferentes según el título (troncales, obligatorias u optativas), se tomó como referencia para este gráfico el título de Ingeniero en Informática por considerarse el más representativo y adecuado en el contexto del presente trabajo

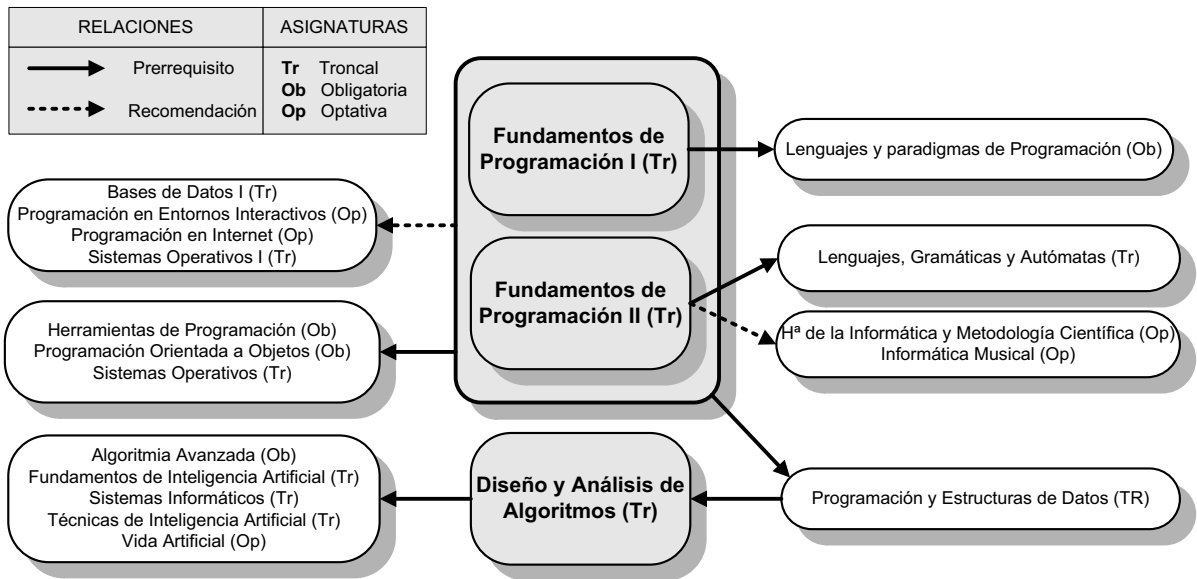


Figura 1.1. Relación existente con las asignaturas de Metodología y Tecnología de la Programación

2. OBJETIVOS, COMPETENCIAS Y PRERREQUISITOS

Este capítulo está dedicado en su primera parte a la definición de los objetivos instrumentales, interpersonales y sistémicos que el alumno deberá ser capaz de alcanzar para completar su aprendizaje de la asignatura y obtener una valoración positiva, haciendo especial hincapié en la construcción de soluciones algorítmicas correctas y eficientes, así como en la capacidad de abstracción necesaria para identificar la solución a problemas nuevos con soluciones a problemas *tipo* bien conocidas.

A continuación se detallan las competencias correspondientes a los perfiles profesionales más relacionados con la materia de la Metodología y Tecnología de la Programación. En el ámbito de la asignatura se especifican las competencias instrumentales, interpersonales y sistémicas que el alumno deberá adquirir con vistas a la consecución de los objetivos descritos. Las competencias instrumentales se centran en la comprensión de conceptos como la corrección y eficiencia temporal y espacial de un algoritmo, así como la aplicación de técnicas de verificación de su corrección por inducción matemática o técnicas de diseño de algoritmos basadas en esquemas. Las destrezas tecnológicas y lingüísticas enumeradas destacan la necesidad de un correcto manejo de las herramientas informáticas y de la notación matemática y algorítmica que permita diseñar e implementar programas correctos, eficientes y robustos.

Entre las competencias interpersonales destacamos la capacidad de realizar una correcta distribución del esfuerzo en las tareas colaborativas, así como el cumplimiento de los plazos definidos en la planificación del trabajo. En cuanto a las competencias sistémicas, la aplicación de los conocimientos y métodos estudiados a problemas del mundo real serán la mejor garantía de que el alumno ha conseguido integrar sus conocimientos teóricos y prácticos, transformándolos en capacidad de afrontar nuevos problemas y obtener soluciones satisfactorias.

Al final del capítulo se describen los prerrequisitos, en forma de competencias y contenidos mínimos cuyo conocimiento previo es necesario para abordar la asignatura. Se detalla asimismo un plan de trabajo para la consecución de dichos requisitos.

2.1 OBJETIVOS

2.1.1 Objetivos instrumentales

- Abordar eficazmente la división de los programas en módulos y especificar adecuadamente los mismos.
- Escribir programas en un tiempo razonable, que funcionen correctamente, que estén bien documentados y que sean legibles.
- Desarrollar soluciones algorítmicas y de tipos de datos ante problemas desconocidos, de un modo riguroso, sistemático y fiable.
- Razonar formalmente sobre la corrección y eficiencia de las soluciones desarrolladas, y compararlas con distintas soluciones alternativas.

- Identificar problemas desconocidos con familias de problemas conocidos y ser capaces de aplicar la solución algorítmica correspondiente.

2.1.2 Objetivos interpersonales

- Fortalecer la responsabilidad en el trabajo a través de la consecución de tareas individuales y en equipo.
- Fomentar la adquisición de habilidades sociales a través de la comunicación entre los miembros de un equipo de trabajo.
- Potenciar el trabajo colaborativo a partir del establecimiento de tareas en equipo que permitan demostrar las capacidades de coordinación, liderazgo y reparto de trabajo.

2.1.3 Objetivos sistémicos

- Integrar conocimientos, métodos y destrezas adquiridos para facilitar la aprehensión de conocimientos en cursos y materias avanzados.
- Evitar planteamientos de "fuerza bruta" en la solución algorítmica a problemas concretos, recurriendo siempre al conjunto de esquemas base estudiados.
- Reforzar la capacidad de abstraer para relacionar problemas nuevos con otros ya conocidos y aplicar soluciones de índole semejante.

2.2 COMPETENCIAS

Siguiendo las directrices marcadas por el Career Space¹, los perfiles profesionales más relacionados con la materia de Metodología y Tecnología de la Programación son:

- Ingeniería de comunicación de datos
- Diseño de redes de comunicación
- Desarrollo de software y aplicaciones
- Arquitectura y diseño de software
- Diseño multimedia
- Asistencia técnica
- Ingeniería de integración, pruebas e implantación
- Especialista en sistemas

La siguiente tabla muestra el conjunto de competencias relacionadas con cada uno de estos perfiles y, por tanto, vinculadas a la materia de Metodología y Tecnología de la Programación.

¹ El Career Space (www.career-space.com) es una iniciativa de nueve importantes compañías europeas de Tecnologías de la Información y las Comunicaciones (BT, Cisco Systems, IBM Europe, Intel, Microsoft Europe, Nokia, Philips Semiconductors, Siemens AG, Thales) y de la Comisión Europea. Una parte de la iniciativa consiste en la definición de perfiles profesionales para las TIC, incluyendo las áreas tecnológicas y tareas asociadas con cada trabajo.

| <i>Perfil</i> | <i>Competencias</i> |
|---|---|
| Ingeniería de comunicación de datos | <ul style="list-style-type: none"> ▪ Diseño, desarrollo, pruebas e integración del software para el nuevo producto. ▪ Ingeniería y resolución de problemas. |
| Diseño de redes de comunicación | <ul style="list-style-type: none"> ▪ Diseñar, construir y poner en funcionamiento prototipos para probar y demostrar distintas funciones. ▪ Ayudar a técnicos de integración y pruebas y trabajar con ellos para que conozcan el diseño. ▪ Detectar y resolver problemas. |
| Desarrollo de software y aplicaciones | <ul style="list-style-type: none"> ▪ Desarrollar el código y los algoritmos de las pruebas o los aspectos de control en tiempo real mediante un procedimiento de trabajo modular que siga la estructura prevista. ▪ Analizar rutinas y módulos, rendimiento, tamaño de memoria, etc. de sistemas técnicos (integrados), cuando proceda. ▪ Prestar apoyo a la dirección del proyecto. ▪ Construir el sistema y los subsistemas de acuerdo con el diseño y la estructura desarrollada y la configuración modular. ▪ Construir prototipos de todo el sistema o de algunas de sus partes. ▪ Colaborar con el arquitecto y el diseñador de sistemas. ▪ Diseñar las pruebas de los módulos y ayudar a diseñar las pruebas de integración e instalación. Realizar la integración del sistema, las pruebas de integración y la instalación. ▪ Desarrollar o aplicar un procedimiento de control de versiones, un procedimiento de instalación y un paquete completo de documentación. Añadir boletines de versión y otros documentos relevantes. ▪ Realizar la presentación técnica, la instalación y las pruebas finales del sistema. |
| Arquitectura y diseño de software | <ul style="list-style-type: none"> ▪ Diseñar soluciones. ▪ Diseñar y probar prototipos. ▪ Definir especificaciones detalladas. |
| Diseño multimedia | <ul style="list-style-type: none"> ▪ Crear prototipos, simulaciones o espacios virtuales para distintas tecnologías multimedia. |
| Asistencia técnica | <ul style="list-style-type: none"> ▪ Instalar, configurar y probar nuevo software operativo, aplicaciones de software y mejoras del software. ▪ Vigilar y mantener sistemas informáticos y redes. |
| Ingeniería de integración, pruebas e implantación | <ul style="list-style-type: none"> ▪ Configurar el producto o sistema para atender las necesidades de los clientes. ▪ Diseñar y realizar pruebas de funcionamiento representativas para demostrar la capacidad. ▪ Asegurarse de que el producto o sistema funciona según las especificaciones. |
| Especialista en sistemas | <ul style="list-style-type: none"> ▪ Proporcionar asesoramiento y orientación respecto el uso, el funcionamiento y el diseño de sistemas o soluciones utilizando productos específicos. |

En lo referente a la asignatura específica objeto de la presente memoria, a continuación se detallan las competencias instrumentales, interpersonales y sistémicas.

2.2.1 Competencias instrumentales

Las competencias instrumentales relativas a la asignatura *Diseño y Análisis de Algoritmos* han sido divididas en habilidades, capacidades y destrezas de tipo cognitivo, metodológico, tecnológico y lingüístico.

Habilidades cognitivas.

- Comprender el concepto de diseño y análisis de algoritmos y sus implicaciones.
- Entender el concepto de eficiencia computacional y distinguir los conceptos de eficiencia en tiempo y en espacio
- Comprender la necesidad de demostrar la corrección de los algoritmos y enmarcar esa necesidad en las técnicas de verificación.
- Conocer la técnica de verificación de algoritmos mediante la inducción matemática.
- Comprender los conceptos elementales sobre la técnica de verificación formal por inducción aplicada a algoritmos iterativos y recursivos.
- Conocer la existencia de familias de esquemas generales de programación que pueden ser utilizados como técnicas básicas de resolución de determinada clase de problemas, así como los tipos de problemas a los que se puede aplicar cada uno de estos esquemas.

Capacidades metodológicas.

- Presentar y justificar las etapas que conducen a la obtención de un buen algoritmo.
- Analizar con rigor la eficiencia de los algoritmos
- Calcular la complejidad temporal o espacial de un algoritmo recursivo o iterativo
- Comparar, respecto a eficiencia, distintas soluciones algorítmicas a un mismo problema
- Instanciar un esquema genérico para un problema concreto a través de la identificación de las estructuras de datos y operaciones propias del problema con las del esquema, previa comprobación de que éstos satisfacen los requerimientos necesarios para poder utilizar el esquema seleccionado.
- Diseñar e implementar pequeñas aplicaciones que requieran varios módulos, utilizando siempre un buen estilo de programación y que enfatizen el esfuerzo intelectual requerido para construir programas correctos y eficientes.
- Justificar la elección de un determinado esquema cuando varios de ellos pueden ser aplicables a un mismo problema.

Destrezas tecnológicas.

- Reforzar los conocimientos adquiridos en etapas previas del aprendizaje sobre los sistemas operativos y lenguajes de programación necesarios para la correcta consecución de las tareas prácticas.
- Manejar con fluidez las herramientas de edición, implementación y compilación de programas para instrumentar los algoritmos de resolución de los problemas planteados.
- Manejo de sistemas electrónicos de consulta documental para la adquisición de conocimiento y resolución de problemas en las fases de implementación.
- Manejo de sistemas electrónicos de búsqueda bibliográfica que complete las sugerencias proporcionadas en el programa de la asignatura.

Destrezas lingüísticas.

- Asegurar el correcto manejo de la notación matemática y algorítmica imprescindibles para cubrir los contenidos de la materia.
- Asegurar la utilización de pseudocódigo para la expresión de algoritmos como lenguaje fundamental de comunicación a lo largo de la docencia de la materia.
- Asegurar el manejo de la terminología específica de la materia que garantice la fluidez en los procesos docentes, tutoriales y de evaluación.

2.2.2 Competencias interpersonales

Responsabilidad en el trabajo y habilidades sociales.

- Asumir roles de responsabilidad parcial o total en el trabajo colaborativo, estableciendo una jerarquía de mando consensuada y asumida por todos los miembros del grupo.
- Cumplir los plazos parciales y totales definidos en la planificación de cada tarea
- Asegurar el conocimiento global de la tarea independientemente del trabajo concreto realizado
- Arbitrar los mecanismos adecuados ante cualquier incidencia que obligue a prescindir de alguno de los miembros del grupo

Trabajo colaborativo.

- Integrar los conocimientos y destrezas desarrolladas en la materia para la consecución de cualquier tarea propuesta en equipo
- Calibrar y determinar un correcto reparto de tareas que garantice un volumen de trabajo equivalente entre los diferentes miembros de un equipo.

2.2.3 Competencias sistémicas

Integración de capacidades cognitivas, destrezas prácticas y disposiciones.

- Aplicación de conocimientos, métodos, esquemas y algoritmos a problemas reales concretos.
- Creación y desarrollo, a partir de los conocimientos adquiridos, de nuevas soluciones ante problemas reales no planteados en el procedimiento docente.

2.3 PRERREQUISITOS

2.3.1 Competencias y contenidos mínimos

- Conocer el concepto de algoritmo y un lenguaje de expresión algorítmica.
- Conocer las estructuras básicas de los lenguajes de programación.
- Conocer las bases matemáticas imprescindibles
- Conocer los conceptos básicos en el uso de las herramientas de edición, programación y compilación.

2.3.2 Plan de trabajo y actividades para la consecución de los objetivos

La docencia anterior a la asignatura *Diseño y Análisis de Algoritmos* materializada en las asignaturas de Fundamentos de Programación I y II debe garantizar la consecución de los prerrequisitos anteriores relativos a la materia de metodología de la programación.

Por otro lado, la base matemática necesaria queda garantizada a través de asignaturas específicas impartidas durante los cursos anteriores que refuerzan las materias correspondientes contenidas en los estudios de segundo grado.

En la clase de presentación de la asignatura, se establecerán con claridad estos prerrequisitos proporcionando, a petición de aquellos alumnos que consideren sus conocimientos insuficientes para cubrirlos, la documentación o bibliografía necesarias para asegurar una mínima homogeneidad en el grupo de trabajo.

3. CONTENIDOS

Este capítulo describe el contenido teórico y práctico de la asignatura, junto a una estimación del tiempo de docencia presencial y prácticas de laboratorio necesario para completar el estudio y asimilación de los contenidos propuestos. Se incluye además una abundante bibliografía básica y complementaria, que resultará útil en la preparación de la asignatura por parte del profesor, o como consulta y ampliación de conocimientos para el alumno.

Las líneas básicas de los contenidos teóricos se sintetizan en tres bloques temáticos principales:

- La eficiencia de los algoritmos
- La corrección de los algoritmos
- La programación con esquemas

Con el desarrollo de estos tres bloques en diferentes unidades temáticas se pretende cubrir los objetivos descritos en el capítulo anterior, permitiendo al alumno adquirir las competencias necesarias para la construcción de programas correctos y eficientes en situaciones de trabajo en grupo en entornos profesionales.

3.1 Contenidos de teoría

Esta sección describe con detalle el contenido teórico de la asignatura. La información referente a cada una de las unidades temáticas refiere a la estructura del tema y el tiempo estimado (en horas) para su desarrollo docente presencial.

El siguiente cuadro presenta la propuesta general de temario de teoría, junto con la distribución temporal de sus contenidos.

TEMA 0: PRESENTACIÓN Y OBJETIVOS DE LA ASIGNATURA (1 hora).

TEMA 1: Introducción al Diseño y Análisis de Algoritmos (1 hora).

1. La resolución de problemas en programación
2. La expresión de algoritmos mediante pseudocódigo

TEMA 2: La eficiencia de los algoritmos (6 horas).

1. Noción de complejidad
2. Cotas de complejidad

3. Notación asintótica
4. Cálculo de complejidades

TEMA 3: La corrección de los algoritmos (4 horas).

1. La corrección de los algoritmos
2. Verificación de algoritmos mediante Inducción

TEMA 4: Programación con esquemas (18 horas).

1. Introducción a la programación con esquemas
2. Divide y vencerás
3. Introducción a la programación dinámica
4. Estrategia voraz
5. Vuelta atrás.

3.1.1 Bibliografía recomendada

FERRI, F.J., J. F. ALBERT y G. MARTÍN (1998) *Introducció a l'anàlisi i disseny d'algorismes*, Universitat de Valencia.

PARBERRY, IAN y WILLIAM GASARCH (2002) *Problems on algorithms. second edition*, <http://hercule.csci.unt.edu/~ian/books/free/>.

BRASSARD, G. y P. BRATLEY (1997) *Fundamentos de algoritmia*.

PEÑA, R. (1998) *Diseño de Programas. Formalismo y Abstracción*, Prentice-Hall.

3.2 CONTENIDOS DE PRÁCTICAS

Las prácticas de laboratorio constituyen una parte importante para la adecuada formación del alumno, ya que les permite conocer y adquirir experiencia en distintas herramientas de programación y les permite abordar el desarrollo de trabajos de cierta complejidad al poner en prácticas los conocimientos teóricos.

Las prácticas deben constituir experiencias concretas que ayuden a los estudiantes a comprender y asimilar los conceptos abstractos adquiridos en las clases de teoría.

Las prácticas estarán relacionadas con los conceptos teóricos fundamentales de la asignatura y se utilizarán para poner en práctica los conocimientos adquiridos en las clases teóricas y facilitar que el alumno asimile dichos conceptos.

Las prácticas se programarán de forma sincronizada con la parte teórica de la asignatura, de manera que el alumno conozca los aspectos teóricos antes de acometer su resolución. Al final de cada práctica, los alumnos elaborarán una memoria que recogerá todos los detalles del desarrollo, resultado, comentarios y conclusiones. Para la realización de esta memoria, los alumnos contarán con una *guía de documentación*, en la que se explica detalladamente los contenidos que tendrán que incluir en la misma. Del mismo modo, se les facilitará una *guía de estilo* para la escritura de los programas, la cual incluye una serie de normas de obligado cumplimiento respecto a la apariencia que deben tener los fuentes de sus programas.

Las prácticas tratarán de ser acumulativas, es decir, cada una servirá para el desarrollo de la siguiente. Esta situación es posible debido al uso de un lenguaje orientado a objetos que permite una fácil reutilización de código.

El siguiente cuadro presenta la propuesta general de temario de prácticas y su distribución temporal.

| <i>PRÁCTICAS</i> | <i>Horas</i> |
|---|--------------|
| 1. Cálculo de complejidades | 10 |
| 2. Esquemas Divide y Vencerás y Programación Dinámica | 10 |
| 3. Esquemas Voraz y Vuelta Atrás | 10 |
| TOTAL | 30 |

3.2.1 Bibliografía recomendada

DEITEL, H.M. y P.J. DEITEL (1995) *Cómo programar en C/C++*. Segunda Edición, Prentice Hall.

JOYANES, L. (2000) *Programación en C++*. Algoritmos, estructuras de datos y objetos., McGraw Hill.

4. CONSIDERACIONES METODOLÓGICAS

En este capítulo se realiza una propuesta de las consideraciones metodológicas aplicables a la asignatura objeto del presente trabajo pero que pueden ser, en la mayoría de los casos, trasladables al resto de las asignaturas de las titulaciones de Informática.

El aprendizaje se puede definir como el proceso de adquisición de conocimientos que explica, en parte, el enriquecimiento y transformación de las estructuras internas de la potencialidad del individuo para comprender y actuar sobre su entorno. El hecho de una transformación de las estructuras internas implica unas estrategias concretas, por parte del docente, que faciliten el desarrollo de un aprendizaje experimental y/o de unas actitudes críticas y creativas en el alumno, es decir, los aspectos formativos de la enseñanza.

Del aprendizaje cabe destacar una serie de características:

- El aprendizaje es personal.
- El elemento clave del mismo es la motivación.
- Es importante que el alumno comprenda la experiencia de aprendizaje.
- La retroinformación es importante para el que aprende.

Por otra parte, la enseñanza se podría definir como el proceso mediante el cual una persona pretende inducir un aprendizaje a otra. Por ello, la enseñanza hace referencia a la actividad del profesor como una tarea cuyo objetivo es el aprendizaje.

En este proceso interactúan unas estrategias cuyo objetivo va más allá del aprendizaje, es decir, la enseñanza no equivale meramente a la pura instrucción academicista, sino que como elemento configurador de la persona, debe reunir los elementos necesarios para inferir unos cambios que complementen los específicos del área objeto de estudio, cambios referidos a abordar los problemas de forma crítica, dotando de funcionalidad cada uno de los aprendizajes que se realizan para que la transferencia entre el aprendizaje que determina el currículo oficial y la sociedad en la que se halla inmerso el universitario se haga de forma creativa.

La forma en que se haya llevado a cabo la enseñanza y el uso que haya hecho el alumno de ese aprendizaje inducido, determinará la eficiencia de la metodología docente empleada.

4.1 PLANTEAMIENTO DOCENTE

Enseñar metodología de programación es todo un reto. Esto es debido principalmente a que el contexto del aula docente es muy distinto del contexto en el que el alumno desarrollará posteriormente su trabajo profesional. La programación en entornos reales de trabajo, en los que la productividad es un factor primordial, puede conducir al alumno a considerar el significado de la siguiente frase como guía de su trabajo:

”El programa funciona y es suficiente.”

La enseñanza de la asignatura está orientada demostrar al alumno lo erróneo de la frase anterior. El método que se va a seguir es la utilización de clases teóricas, para transmitir los conceptos básicos; y las clases prácticas, para asentar los conocimientos adquiridos en las clases teóricas y comprobar su utilidad mediante la resolución de ejercicios.

Modelo de Teoría.

En las clases de teoría, el profesor hará uso de ejemplos que ayuden a los alumnos a entender los conceptos teóricos expuestos. Los ejemplos deben facilitar la aplicación de dichos conceptos a casos más complejos.

La exposición oral de las clases teóricas se complementará con la proyección de transparencias y el uso de la pizarra. Además, los alumnos disponen en la Biblioteca de la UA de la bibliografía básica y complementaria que se detallará posteriormente. Por último, cabe señalar que se hace un uso extensivo del Sistema de Información integrado de la UA para facilitar material (copia de transparencias, ejercicios resueltos, etc.) a los alumnos.

Modelo de Prácticas.

Dadas las características de estas asignaturas, las prácticas de laboratorio van a jugar un papel muy importante para la adecuada formación del alumno. En primer lugar, les debe permitir conocer y adquirir experiencia en distintas herramientas de programación y en segundo lugar, deben facilitar que el alumno asimile y experimente los contenidos teóricos de las signaturas.

El papel que el profesor desempeñará no se limita a asistir al alumno en la aclaración y resolución de dudas particulares referentes a la práctica, sino que se propone un modelo de interacción con el alumno que amplían las funciones del profesor. Para cada práctica, el profesor presentará su planteamiento, justificación y objetivos, así como una planificación temporal de tareas intermedias a realizar que permitan al alumno una correcta organización de su trabajo. Aprovechando esta organización se efectuarán discusiones en grupo para abordar los contenidos y métodos a utilizar en cada tarea intermedia.

Este planteamiento permite que los alumnos se involucren activamente en el desarrollo de la práctica planteando, comentando, analizando y comparando las ventajas e inconvenientes de las propuestas presentadas por el profesor en las prácticas y las suyas propias. A su vez, la mayor interacción alumno-profesor facilita una mejor comprensión de los conceptos introducidos en las clases teóricas y un buen seguimiento del trabajo de los alumnos por parte del profesor.

4.2 ESTRATEGIAS DE APRENDIZAJE

El método propuesto, aunque inspirado en el modelo constructivista del aprendizaje, tiene en consideración las limitaciones estructurales del contexto masificado en el cual se desarrolla la actividad docente. Por ello se propone el modelo mixto de lección magistral con aprendizaje activo.

Así, las propuestas metodológicas planteadas se basarán en el establecimiento de los objetivos de cada asignatura, así como una serie de actividades didácticas dirigidas a la adquisición de conocimientos mediante las clases teóricas y de problemas, las clases de prácticas en el laboratorio, el proceso tutorial y una serie de actividades adicionales.

4.2.1 El establecimiento de objetivos

En una adecuada metodología didáctica intervienen fundamentalmente tres factores: los objetivos propuestos, los medios disponibles y las posibilidades de los sujetos que participan en el acto educativo.

El primer factor vendrá definido por los planes de estudio, y matizado por la labor docente del profesor. Así la tarea del docente será la de establecer los medios en función de los objetivos y los objetivos en función de los medios, y ambos conceptos con las posibilidades reales que la situación genera.

De este conjunto de relaciones se extraerá la propuesta de objetivos que el docente planifica para la asignatura. Un objetivo debe indicar, sin posibles ambigüedades, cuáles son las tareas que debe realizar el alumno y con qué contenidos.

A medida que estas tareas se concretan, los objetivos se clarifican pasando de objetivos generales a objetivos específicos. Es muy importante que en este proceso de concreción de los objetivos se conserve la coherencia, de forma que no se pierdan los objetivos generales de la asignatura al concentrarse en los detalles de la misma. Una presentación de los objetivos bien realizada puede ayudar a conseguir una motivación en el alumno que le facilite superar muchos obstáculos posteriores.

4.2.2 La adquisición de los conocimientos

De acuerdo con los objetivos generales y con la estructuración de los contenidos, se indicarán los métodos, procedimientos e instrumentos que, de forma genérica, se van a utilizar.

Las clases de teoría. En estas clases se presentarán los contenidos básicos de un cierto tema. Comienzan por una introducción donde se comentarán los contenidos del tema y sus objetivos específicos, así como un breve comentario sobre los conceptos ya vistos en clases anteriores y que tengan incidencia en los puntos a tratar.

A continuación se realizará la exposición de los contenidos, haciendo uso de aquellos medios audiovisuales, textos y documentos que permitan un adecuado nivel de motivación e interés en los alumnos.

En la metodología propuesta se compaginarán clases expositivas-frontales (de tipo magistral) con clases expositivas-interactivas (solución de problemas, estudio de casos prácticos, trabajos por grupos, etc.), intentando hacer uso de estas segundas siempre que sea posible. Las características de las asignaturas propuestas facilitan dicha aproximación ya que se pueden introducir

los conceptos básicos para después pasar a proponer y solucionar problemas basados en dichos conceptos básicos.

En esa línea, se debe motivar a los alumnos a intervenir en cualquier momento en las clases de teoría, exponiendo sus dudas, realizando comentarios que se intentarán aprovechar para realizar de forma más dinámica la clase y facilitar el aprendizaje deseado por parte de los alumnos.

Por último, es interesante dedicar siempre un periodo de la clase para mostrar las conclusiones más importantes a modo de resumen del tema.

Las clases de prácticas. Las prácticas son también un medio excelente para que el alumno potencie su iniciativa y capacidad crítica. Además, son imprescindibles para conseguir un buen aprendizaje en materias científicas y tecnológicas, permitiendo proyectar los conocimientos acumulados sobre problemas reales. Por todos estos motivos es necesario prestar a las prácticas una gran atención y dedicación, para conseguir hacerlas útiles y sugestivas. Así, la planificación y supervisión deben ser completas y directas.

Por último, las clases prácticas permiten al profesor estar en mayor contacto con los alumnos debido al número reducido de alumnos por cada grupo, permitiéndole realizar un mejor seguimiento de cada alumno.

El enfoque utilizado en las prácticas es el siguiente:

- Se plantean como una serie de problemas de complejidad creciente.
- Se realizará un seguimiento continuado a las tareas que realiza el alumno. Este seguimiento permite observar los progresos que realiza el alumno a lo largo del curso. Además, se pretende evitar que el alumno inicie sus prácticas con planteamientos incorrectos. Por último, se establece la comunicación con el alumno que permite que se le puedan indicar los errores que está cometiendo y el modo de solucionarlos.
- Se deberán realizar en parejas. Con ello se pretende fomentar el trabajo en equipo y favorecer el intercambio de conocimientos entre los integrantes de la pareja (desarrollo de habilidades de comunicación tanto oral como escrita). Sin embargo, se especificará que la evaluación de sus conocimientos se realizará de forma individual.
- Los alumnos dispondrán con suficiente antelación de los enunciados de las prácticas. Los enunciados deberán contener: objetivos generales, materias de trabajo, contenido mínimo de las memorias que se exigirán en cada práctica y fecha de entrega.
- Deberán estar sincronizadas con las clases de teoría. Así se consigue que el alumno conozca los aspectos teóricos previamente al planteamiento del trabajo práctico.

El proceso tutorial. Las horas de asistencia al alumnado o tutorías son esenciales para la calidad de la enseñanza. Estas horas son la continuación de las clases de exposición, de problemas y prácticas de laboratorio.

La atención personalizada en las horas de asistencia al alumnado mejora el aprendizaje y el conocimiento entre profesor y alumno. Los alumnos podrán acudir al profesor dentro de un horario estipulado.

Desgraciadamente, un elevado número de alumnos no acude a tutorías o hace un escaso uso de ellas. Probablemente, ello se debe a que los alumnos no estudian de forma continuada y sólo acuden al profesor en las vísperas del examen o de la entrega de trabajos. Por ello, es muy importante recordarles que disponen de ellas durante todo el curso y animarlos a que acudan.

Los objetivos de las tutorías son:

- Motivar al alumno en la asignatura.
- Informar al alumno sobre contenidos teórico-prácticos de la misma.
- Ayudar al alumno en la resolución de los ejercicios propuestos en teoría y prácticas.

Las actividades adicionales. Es conveniente que el alumno pueda realizar algunas actividades adicionales que le estimulen e incrementen su interés por la asignatura. Estas actividades deben ser optativas, ya que su principal objetivo es permitir al alumno profundizar en algunos temas y no marcar un requisito adicional para superar la asignatura. Las actividades pueden consistir en asistencia a seminarios y conferencias de interés sobre temas puntuales o bien realizar trabajos de curso que potencien el trabajo en equipo.

Así mismo, la participación por parte del profesorado en redes de investigación docente fomenta la búsqueda de nuevos espacios, técnicas y metodologías de trabajo para fomentar las mencionadas habilidades.

Los recursos de Internet. Las asignaturas ofrecen páginas web a través de las cuales los alumnos pueden adquirir el material necesario para la docencia tanto práctica como teórica, consultar los objetivos, temario y planteamiento de las asignaturas, así como estar al tanto de sus noticias y aspectos dinámicos.

Además, el correo electrónico proporciona una excepcional vía de comunicación entre profesores y alumnos al permitir resolver ciertas dudas puntuales sin necesidad de producirse un contacto presencial.

Por último, destacar que la Universidad de Alicante proporciona un Campus Virtual a través del cual se proporciona un servicio de complemento a la docencia y a la gestión académica dirigido tanto al profesorado como al alumnado y al personal de administración de la Universidad de Alicante.

Entre otras funcionalidades se encuentran:

- Publicación de la bibliografía.
- Publicación enlaces de interés.
- Tutorías electrónicas.

- Publicación de un repertorio de preguntas frecuentes.
- Propuesta y moderación de debates.
- Publicación de controles y sus notas.
- Suministro de materiales docentes.
- Listas de clase.
- Fichas de alumnos.
- Información sobre planes de estudio.
- Gestión de anuncios de interés.
- Publicación de horarios de tutorías presenciales.

4.3 EL PROCESO DE EVALUACIÓN

Como consecuencia del método docente planteado, la evaluación tendrá como misión fundamental cuantificar el grado de cumplimiento de los objetivos, además debe ser una contribución esencial al proceso formativo de los alumnos en forma de seguimiento y realimentación que valore sus esfuerzos y progresos y les ayude a aceptar el error como un elemento más del proceso de aprendizaje.

Considerada como realimentación del método docente, nos permitirá analizar el grado de cumplimiento de los objetivos, valorar el progreso de cada alumno y realizar un estudio de la eficacia del método docente.

Toda evaluación debe partir de los siguientes supuestos:

- Debe responder a normas predefinidas y conocidas de antemano por todos los profesores y alumnos.
- Debe ser coherente con los objetivos fijados.
- Debe ser objetiva y fiable.
- Debe abarcar todos los niveles de conocimiento y toda actividad del alumno con relación a la materia estudiada.
- Debe realizarse de forma continuada para facilitar las correcciones necesarias, tanto por parte del alumno, como del profesor.

Aunque sería lo deseable, el último supuesto no siempre es viable en las asignaturas objeto del presente proyecto debido al elevado número de alumnos matriculados tanto en las clases prácticas como en las teóricas. No obstante, es posible establecer pequeños controles que permitan extender la valoración del alumno hacia un conjunto de aspectos del proceso educativo como pequeños ejercicios escritos, tutorías, actividades voluntarias, etc.

Los exámenes serán pruebas escritas que, aunque han sido criticadas por diversos motivos, como que inducen al alumno a estudiar con el único fin de superar el examen o que de él se obtiene información escasa, siguen siendo la forma de evaluación más viable. Para intentar conseguir que estos exámenes sirvan mejor al proceso de aprendizaje debería cumplirse que:

- El examen sea una revisión global de la materia considerada.
- El examen sea corregido lo antes posible, discutiéndose las posibles respuestas, los errores que han aparecido, etc. Para ello se puede dejar una solución por escrito a disposición del alumno donde se realicen los comentarios sobre el examen que el profesor considere oportuno.
- Las revisiones de los exámenes no se deben reducir a discutir para obtener la calificación necesaria para aprobar, sino que además de subsanar posibles errores cometidos en la corrección, se debe comentar al alumno los motivos de la calificación del examen.

Además de los exámenes escritos, el otro elemento fundamental en la evaluación son las prácticas de laboratorio. Estas prácticas se valorarán atendiendo a las memorias que los alumnos entregarán y a la defensa que realicen de las mismas.

4.3.1 Criterios de evaluación

Las notas finales se expresarán de forma numérica dentro de los siguientes rangos:

- Suspenso (0-4,99)
 - El conocimiento y la comprensión del contenido cubierto en el curso ha sido insuficiente
 - Los problemas planteados han sido resueltos de forma inadecuada
 - Los resultados obtenidos en las fases prácticas no han sido correctamente entendidos
 - Las prácticas se han desarrollado incorrectamente
 - La participación en las clases ha sido nula
- Aprobado (5-6,99)
 - El conocimiento y la comprensión del contenido cubierto en el curso ha sido correcto
 - Los problemas planteados han sido resueltos de forma adecuada
 - Aunque con algunos problemas, los resultados obtenidos en las fases prácticas han sido entendidos
 - Las prácticas se han desarrollado bajo mínimos pero adecuadamente
 - La participación en las clases ha sido escasa
- Notable (7-8,99)
 - El conocimiento y la comprensión del contenido cubierto en el curso ha sido adecuado

- Los problemas planteados han sido resueltos satisfactoriamente
- Las prácticas se han desarrollado adecuadamente y los resultados han sido correctamente comprendidos
- La participación en las clases ha sido activa
- Sobresaliente (9-10)
 - El conocimiento y la comprensión del contenido cubierto en el curso ha sido altamente satisfactorio y ampliado más allá del programa por la propia iniciativa del alumno
 - Los problemas planteados han sido resueltos con destreza y demuestran un claro interés y habilidad en la materia desarrollada
 - El grado de experimentación del alumno denota una gran capacidad de análisis y comprensión de los resultados
 - La participación en las clases ha sido elevada y el alumno ha puesto de manifiesto un interés especial que le ha llevado a tomar la iniciativa en propuestas propias durante el curso

Opcionalmente, y dependiendo de las circunstancias del grupo y del curso, el profesor podrá asignar las correspondientes *matrículas de honor* a aquellos alumnos que hayan sido calificados con las notas máximas (sobresalientes) y hayan destacado por alguna especialmente.

5. PLAN DE TRABAJO DE LOS ALUMNOS

Los siguientes cuadros especifican el tiempo y el esfuerzo necesarios por parte del alumno para cada una de las diferentes unidades temáticas incluidas en la asignatura *Diseño y Análisis de Algoritmos*¹.

NÚMERO DE HORAS PRESENCIALES

| <i>Unidad temática</i> | <i>Teoría</i> | <i>Práctica</i> |
|---|---------------|-----------------|
| TEMA 0: Presentación | 1 | |
| TEMA 1: Introducción al Diseño y Análisis de Algoritmos | 1 | 2 |
| TEMA 2: La eficiencia de los algoritmos | 6 | 8 |
| TEMA 3: La corrección de los algoritmos | 4 | 2 |
| TEMA 4: Programación con esquemas | 18 | 18 |
| Preparación al examen final | 2 | |
| Examen final | 2 | |
| TOTAL: 64 | 34 | 30 |

NÚMERO DE HORAS NO PRESENCIALES

| <i>Unidad temática</i> | <i>Teoría</i> | <i>Práctica</i> |
|---|---------------|-----------------|
| TEMA 1: Introducción al Diseño y Análisis de Algoritmos | 1,5 | 3 |
| TEMA 2: La eficiencia de los algoritmos | 9 | 12 |
| TEMA 3: La corrección de los algoritmos | 6 | 3 |
| TEMA 4: Programación con esquemas | 27 | 27 |
| Preparación al examen final | 6 | |
| Tutorías | 3 | 3 |
| Foros virtuales | 3 | 3 |
| TOTAL: 106,5 | 55,5 | 51 |

¹ Las horas no presenciales se corresponden con la recopilación y estudio de información complementaria a los trabajos presenciales así como a la participación en tutorías y foros virtuales.

BIBLIOGRAFÍA

BRASSARD, G. y P. BRATLEY (1997) *Fundamentos de Algoritmia*, Prentice Hall.

DEITEL, H.M. y P.J. DEITEL (1995) *Cómo programar en C/C++*. Segunda Edición, Prentice Hall.

FERRI, F.J., J. F. ALBERT y G. MARTÍN (1998) *Introducció a l'anàlisi i disseny d'algorismes*, Universitat de València.

JOYANES, L. (2000) *Programación en C++*. Algoritmos, estructuras de datos y objetos, McGraw Hill.

KNUTH, D.E. (1973) *The art of Computer Programming*.; Volume 3: *Sorting and Searching*, Addison Wesley.

PARBERRY, IAN y WILLIAM GASARCH (2002) *Problems on algorithms*. second edition, <http://hercule.csci.unt.edu/~ian/books/free/>.

PEÑA, R. (1998) *Diseño de Programas*. Formalismo y Abstracción, Prentice-Hall.