

Las asignaturas de Ciencias de la Computación en la Escuela Universitaria de Ingeniería Técnica Industrial de Madrid

Carlos Platero, Miguel Hernando, Roque Salterán, Alfonso Reimóndez, Roberto González
Escuela Universitaria de Ingeniería Técnica Industrial (UPM), c/ Ronda de Valencia, 3, 28012 Madrid,
cplatero, mhernando, rsaltaren, areimondez, rgonzalez @elai.upm.es

Resumen

Las asignaturas de Ciencias de la Computación de la EUITI-UPM se están dando desde el Área de Conocimiento de Ingeniería de Sistemas y Automática. Cuatro asignaturas son dadas: Fundamentos de Informática, Informática Industrial, Redes de Computadores e Ingeniería de la Programación. En esta ponencia se desarrollan sus enfoques, sus temarios, las prácticas y los criterios de evaluación emprendidos en estos años.

1 INTRODUCCIÓN

El nuevo Plan de Estudios de la EUITI-UPM fue publicado en el BOE con fecha 24 de mayo de 2001. En el curso 2002/03 se dio los primeros paso de esta nueva andadura. El Área de Ingeniería de Sistemas y Automática, en la EUITI-UPM, tiene vinculada las imparticiones de las asignaturas troncales de Fundamentos de Informática e Informática Industrial, junto con la asignatura obligatoria de universidad de Redes de Computadores y la optativa de Ingeniería de la Programación.

Aunque para los planes de Ingeniería Técnica Industrial, la asignatura de Fundamentos de Informática no está asociada al Área, en la EUITI-UPM al carecer de ninguna de las áreas de conocimiento publicas en las directrices propias de las titulaciones de Ingeniería Técnica Industrial, se asocio a la nuestra. Por acuerdo de la Junta de Escuela, la asignatura de Fundamentos de Informática de las cuatro titulaciones, impartidas en la EUITI-UPM (Mecánica, Química, Electricidad y Electrónica), fue asociado al área ISA.

La asignatura de Informática Industrial se dio por primera vez en el curso 2003/04. Esta asignatura troncal que aparece en la titulación de Electrónica Industrial si está asociada al área ISA.

Por último, las asignaturas de Redes de Computadores y de Ingeniería de la Programación son materias de último curso. La primera es común a todas las intensificaciones y la otra, asignatura optativa, está ligada a la intensificación de

Automatización Avanzada, en la titulación de Electrónica Industrial.

La ponencia se organizada dando los aspectos más relevantes de las cuatro asignaturas en los siguientes apartados.

2 FUNDAMENTOS DE INFORMÁTICA

La asignatura de Fundamentos de Informática, se imparte en el primer año, a todas la titulaciones de Ingeniería Técnica Industrial de la EUITI-UPM. Esta asignatura, se imparte durante un cuatrimestre con tres horas de carga teórica por semana y un total de 10 horas practicas para todo el cuatrimestre. De acuerdo a las horas asignadas se puede ver que hay una descompensación entre la teoría y la practica (laboratorio), esto afecta a un numero importante de estudiantes que aunque aprueban, carecen de la experiencia suficiente en programación, que sólo se puede adquirir con un número suficiente de horas practicas. No es difícil deducir que estos aspectos de la docencia, generan problemas en asignaturas más avanzadas. Para paliar estas carencias formativas es necesario insistir en la necesidad de disponer de suficientes medios docentes y de infraestructura. Aparte de cualquier consideración, las asignaturas donde se enseñan lenguajes de programación, deben ser enfocadas a una permanente interacción profesor-estudiante. Por lo que idealmente cada estudiante debería en clase, disponer de un ordenador para editar y ejecutar en clase los procedimientos que dan lugar al aprendizaje correcto de la sintaxis, la lógica y la algorítmica de un lenguaje. Sin duda, esto es más relevante que el programa de la asignatura, el cual fácilmente se puede estructurar siguiendo el orden de un libro de texto, suficientemente conocido. En esta Escuela después de varias discusiones se ha optado usar como libro de texto, un libro de un autor muy conocido y que periódicamente es revisado y actualizado [1]. El programa del curso se basa en los primeros capítulos (1 a 8) del libro referenciado, dejándose los temas de manejo dinámico de listas y otros para ser tratados en cursos mas avanzados, dependiendo de la especialidad. En base a uno de los criterios del organismo ABET [2], es esencial definir un libro de texto para cada asignatura, el cual debe unificar el contenido de la docencia por parte de los profesores de una asignatura como Fundamentos de

Informática, que requiere en la EUITI-UPM de al menos seis docentes. Un libro de texto, permite también disponer de problemas suficientes al final de cada capítulo, con el fin de conducir el trabajo en casa de los estudiantes. Varios criterios de ABET, no son posibles de aplicar en nuestro caso, uno de ellos por ejemplo, es la realización de un número suficiente de exámenes parciales y un examen final (estos exámenes permiten controlar la formación de los estudiantes más periódicamente y esta demostrado que dan lugar a una evaluación más justa y objetiva), otra vez por la carencia de medios docentes, afectados por la masificación de las asignaturas y por la estructura de evaluación de nuestras universidades. Otro criterio importante para ABET, es la dependencia de la matriculación de una asignatura, de los prerrequisitos (aprobados o vistos) Los prerrequisitos obligatorios de una asignatura, hacen más “fuerte” las exigencias de conocimientos previos de formación de los estudiantes y validan el éxito de la experiencias formativas anteriores y garantizan una base mínima y la calidad de los resultados de una asignatura, este es un tema polémico, cuya discusión debe abordarse formalmente tarde o temprano, para poder medir y garantizar la calidad de la docencia y la formación rigurosa de los estudiantes.

3 INFORMÁTICA INDUSTRIAL

En las directrices propias de la titulación de Electrónica Industrial aparece la asignatura de Informática Industrial, vinculada al Área. Su carga es de 9 créditos troncales, al que se le añadió 1.5 créditos adicionales en la EUITI-UPM.

En la breve descripción del contenido aparece el párrafo “*El microprocesador y el computador en el control de procesos*”. Cuando se empezó a elaborar el programa, se tenía claro que en la ordenación temporal, los alumnos ya habían recibido, en el curso pasado, la asignatura de Fundamentos de Informática. De hecho, en la EUITI-UPM Fundamentos se da en el primer curso, mientras Informática Industrial se imparte en el segundo curso, concretamente, en el cuarto cuatrimestre.

Bajo estas circunstancias, los alumnos que reciben la asignatura de Informática Industrial, ya tienen conocimientos de programación estructurada (lenguaje C), de arquitectura de computadores y de sistemas operativos.

Con estos prerrequisitos se construyó el temario de la asignatura. Entendíamos que el siguiente paso sería una asignatura sobre el paradigma de la Programación Orientada a Objetos (POO). Con esta perspectiva, se decidió que el marco teórico tendría

una carga de 6 créditos, los problemas de 1.5 créditos y las prácticas de laboratorio de 3 créditos.

Al pensar en un curso de POO, se decidió que en la teoría debería de desarrollarse, por un lado, un lenguaje de POO y de otro una introducción al análisis y diseño orientado a objetos (AOD) y al Proceso Unificado (UP). Como dice Larman[3] “tener un martillo no te hace un arquitecto”, esto es especialmente cierto en la programación orientada a objetos. Conocer un lenguaje OO, no supone pensar en un sistema de objetos.

Teniendo la filosofía de la asignatura, ésta se procedió a presentarla con 3 horas/semana de teoría-problemas sobre C++ y 2 horas/semana a la teoría-problemas al AOD y al UP. Respecto a las prácticas se decidió dar 10 prácticas de 2 horas cada una (1 práctica por semana), de las cuales ocho están dedicadas a C++ y dos sobre ingeniería inversa y patrones GRASP y GoF[4].

El temario se confeccionó sobre los pilares que se requiere para la POO, cuyas palabras claves aparece en la figura 1.

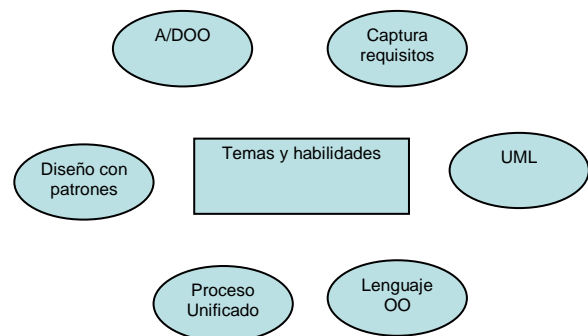


Figura 1 Habilidades en el paradigma orientado a objetos

Para concretar los conceptos teóricos manejados en la teoría de la POO se utiliza el lenguaje C++. Como el énfasis del curso se pone en la transmisión de estos conceptos más generales, y desligados de la realización específica en un determinado lenguaje, el temario resalta los elementos particulares de la POO y hace poco hincapié en otros aspectos tradicionalmente considerados como características de este lenguaje. El 50 % del tiempo asignado a esta parte del temario se centra en explicar los conceptos de encapsulamiento, herencia y polimorfismo. En este caso se entra en detalle a explicar los mecanismos con los que C++ resuelve estos aspectos. Sin embargo, se da menor peso al concepto de clases y funciones genéricas o a las funciones de entrada y salida estándar. Estos dos temas destacan por ser tradicionales en los cursos de C++, sin embargo tienen un pobre reflejo en la teoría general de POO.

El 20% de las horas son inicialmente utilizadas para refrescar los conceptos y la sintaxis de C bajo el pretexto de explicar las denominadas modificaciones menores del lenguaje. La experiencia de proceder de esta manera ha resultado positiva puesto que un alumno que ha pasado la asignatura de Fundamentos de Informática habitualmente ha olvidado muchos de los conceptos manejados como consecuencia de no programar durante los meses de verano.

Para adaptar el curso al enfoque indicado se han editado unos apuntes electrónicos de esta parte del curso que se han ido publicando según se finalizaban los temas, de forma que el alumnos se sentía obligado a tomar apuntes durante las sesiones teóricas. Actualmente estos apuntes están disponibles en la página web del departamento. La tabla muestra las horas utilizadas en el desarrollo del temario del curso de C++.

Tabla 1 Temario de POO

Tema	Contenidos	Horas
Introducción a C++	Introducción a la OOP, evolución del C : ANSI C, C++ y C#.	3H
Modificaciones menores al lenguaje C	Modificaciones en la sintaxis. Variables y especificadores. Sobrecarga de funciones. Operadores new y delete. Entradas y Salidas.	8H
Clases Objetos y métodos	Conceptos básicos. Secciones públicas y privadas. Constructores y destructores. New y delete con clases. Clases y funciones friend. Sobrecarga de operadores. Variables y funciones miembro static.	10H
Herencia	Definición. Herencia simple y múltiple. Clases virtuales. Conversión entre objetos.	6H
Polimorfismo	Funciones virtuales. Funciones virtuales puras. Clases abstractas. Destructores virtuales.	5H
Plantillas	Concepto. Plantillas de funciones. Plantillas de clases	5H
Excepciones	errores y excepciones en C++, throw, try.	3H

En la parte de introducción al AOO/D y al UP, el alumno debe de conocer las disciplinas y artefactos que le permita desarrollar un sistema orientado a objetos. En este sentido, el temario parte de una introducción al Proceso Unificado, posteriormente se le expone los artefactos que presenta UP para la captura de requisitos, siguiendo los moldes de Cockburn[5].

Se sigue con la etapa de análisis, introduciéndose en el modelado del dominio. Presentando las técnicas de captura de las clases conceptuales y de las asociaciones, para crear el diccionario visual del

problema. Por tanto, en este momento, se hace factible introducir UML. Primeramente, se hace ver UML estático y posteriormente se pasa al dinámico.

El último tramo y al que se le dedica más tiempo es a la etapa del diseño. El alumno ya tiene todas las herramientas de notación, UML, y ha pasado por las etapas de recogida y modelado del dominio. Ahora, con todos estos artefactos, se pasa a desarrollar el diseño con patrones de aprendizaje GRASP. Los diagramas de interacción y el reparto de responsabilidades, siguiendo los patrones, permite al alumno vislumbrar la construcción del sistema orientado a objetos, desapareciendo en gran medida la subjetividad del diseñador. Para acabar, se dan algunos patrones GoF y se explica, dentro del modelo de implementación, el método de Programación Extrema [6].

Un resumen de los descriptores del temario y del reparto de horas es presentado en la tabla 2.

Tabla 2 Temario y dedicación a la introducción AOO/D (3 créditos)

Tema	Contenidos	Horas
Introducción a la Ingeniería del SW	El paradigma orientado a objetos. El Proceso Unificado	3H
Recogida y documentos de	Visión y glosario. Casos de Uso. Especificaciones complementarias	4H
AOO	Diagramas de secuencias del sistema. Modelo del dominio. Los contratos de operación.	3H
UML estático	Clases y conceptos afines. Representación de los objetos. Tipos de relaciones entre clases.	5H
UML dinámico	Los diagramas de interacción. El diagrama de estado y de actividades. Los diagramas de implementación.	3H
DOO	Patrones GRASP. Realización de casos de uso con los patrones GRASP. Modelo de diseño: creación de los diagramas de clase. Más patrones GRASP. Diseño con patrones GoF. Diseño de la arquitectura lógica con patrones. Modelo de implementación.	12H

El diseño del temario de prácticas tiene unas particularidades notables. Debe intentar mostrar de forma experimental los conceptos de POO a través de C++ y reflejar además los conceptos manejados en AOO/D y UP. Por otro, las prácticas se diseñaron con el fin de motivar y dar herramientas al alumno de cara a la realización del trabajo obligatorio para pasar la asignatura. Es por ello que las prácticas se han diseñado de forma escalonada, de forma que el

trabajo realizado en prácticas anteriores sirvan para lograr objetivos más ambiciosos en las prácticas posteriores. La tabla muestra las practicas realizadas durante el curso 03/04.

Las sesiones prácticas se imparten en el aula de informática del departamento, compuesta por 23 puestos. Los grupos de laboratorio constan de 14 o 15 alumnos máximo, ocupando un alumno cada puesto.

Tabla 3 Prácticas de Informática Industrial

Número de la práctica	Título	Conceptos
1	La clase CString	Introducción al concepto de objeto
2, 3	Encapsulado y sobrecarga: La clase complejos	Durante dos prácticas se desarrolla una clase para manejar números complejos.
4	La herencia: La clase biblioteca	Se muestra la reutilización de código gracias al mecanismo de herencia
5	Introducción a Rational Rose	Ingeniería directa del software
6	Programación de un diálogo	Introducción a la programación sobre windows haciendo uso de la POO.
7	Gráficos 3D	Continuación de la programación en windows y uso de OpenGL
8	Animación y Polimorfismo	Utilizando las dos prácticas anteriores y el mecanismo de polimorfismo se realiza un a animación tridimensional de un sistema planetario.
9	Patrón GoF. Observador	Ingeniería Inversa del Software.

En cuanto a la evaluación de la asignatura, la nota del examen sólo representa el 50% del total. El resto proviene del trabajo de curso y del examen de prácticas.

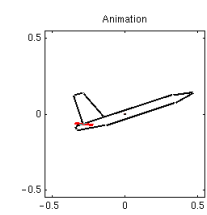
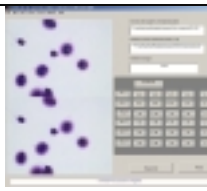
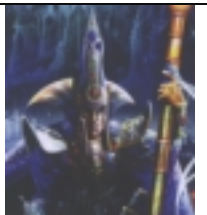
El trabajo de curso pretende ser un proceso de síntesis, donde los estudiantes pongan, sobre un caso práctico, las habilidades y los artefactos dados en las clases. Los alumnos forman grupos de dos y se les da una problemática concreta, solicitándoles que entreguen los siguientes documentos:

1. Documentación de los requisitos: visión del proyecto, casos de uso, glosario y las especificaciones complementarias.
2. Modelo del dominio y los contratos de operaciones más importantes.
3. Los diagramas de interacción.

4. Diseños de las clases según patrones (GRASP y GoF) y con arquitectura en capas.
5. Modelo de Implementación

En la tabla 4 se muestran algunos ejemplos de trabajos mandados durante el curso 2003/04.

Tabla 4 Ejemplos de trabajos de curso de Informática Industrial (curso 03/04)

Título	breve descripción	Imagen
Simulador de Piloto de un avión.	Se pide simulación del paso a paso de la aeronave ante una petición del piloto. También se puede modificar las estrategias de control.	
Análisis de imágenes biomédicas	Realizar tareas de procesado, segmentación y clasificación de imágenes biomédicas proveniente de una etapa de adquisición con microscopía.	
Juego Archon EUTI	Diseño de juego empleando OpenGL	

Los exámenes están constituido por un cuestionario de preguntas teóricas de tipo test, una pregunta de tema a desarrollar, varios ejercicios de ejecución e interpretación de código y un problema global donde se entronca el Proceso Unificado. En éste se da un enunciado y debe de rellenarse un caso de uso, realizar el modelo del dominio, los diagramas de secuencias del sistema, los contratos de operación, los diagramas de interacción, la determinación de las clases de diseño y el modelo de implementación en C++.

3 REDES DE COMPUTADORES

La asignatura de Redes de Computadores, común a todas las intensificaciones de la titulación de Electrónica Industrial, tiene una carga de 4,5 créditos obligatorios y se imparte en el sexto cuatrimestre.

En un mundo en el que las comunicaciones tienen cada vez más importancia, las redes de ordenadores se están convirtiendo en una "utility", de forma que ya nadie se plantea una oficina sin red de

computadores como no se plantea una casa particular sin teléfono. Esta integración de las redes de computadores también afecta a los procesos industriales donde se aprovecha su potencia en diversas aplicaciones. Por este motivo, resulta de especial interés completar la formación de la titulación de Electrónica Industrial con los aspectos fundamentales de las redes de computadores.

El objetivo fundamental de esta asignatura es transmitir al alumno los conceptos fundamentales de las redes de computadores, partiendo de los fundamentos teóricos planteados sobre el modelo OSI para centrarse posteriormente en el modelo TCP/IP. Dado que se trata de una titulación técnica se abordan en mayor profundidad los aspectos prácticos de la materia haciendo un repaso más general sobre los aspectos teóricos.

En su parte teórica la asignatura se estructura en base a cuatro grandes bloques:

1. Fundamentos de redes
2. Arquitectura TCP/IP
3. Capa de aplicaciones
4. Redes WAN.

En el primer bloque, denominado fundamentos de Redes, se describen conceptos básicos de comunicaciones sin entrar en profundidad, tales como transmisión, sincronización, multiplexado, el modelo OSI, etc. En el segundo bloque, se describe la arquitectura TCP/IP, es decir, conceptos fundamentales como, direccionamiento IP, protocolos de enrutamiento, protocolos de transporte, filtros, etc. En el tercer bloque se da una pincelada sobre los servicios más importante correspondientes al nivel de aplicación como son el servicio web, servicios de transferencia de ficheros(FTP), resolución de nombres de dominio (DNS), etc. Finalmente, en el cuarto bloque dedicado a la redes WAN, se describen los protocolos más comúnmente empleados en redes de área extensa, tales como Frame Relay.

En su parte práctica, la asignatura proporciona las nociones básicas necesarias para configurar una red elemental de comunicaciones. En un primer paso se muestra al alumno los mecanismos de configuración de red en un computador, usuario de la infraestructura de red. A continuación se muestran como configurar los equipos que conforman una red de comunicaciones básica que garantice la conexión de una serie de computadores en un área local. Y finalmente, se muestra la forma de conectar una red local a una red de área extensa. Como recurso para las clases prácticas, además de los computadores, se emplea una aplicación de simulación de redes.

4 INGENIERÍA DE LA PROGRAMACIÓN

Esta asignatura optativa se encuentra encuadrada en la intensificación de Automatización Avanzada de la titulación de Electrónica Industrial. El bloque de intensificación tiene una carga de 24.5 créditos, repartido en tres asignaturas de 7.5 créditos. Junto con Ingeniería de la Programación se da Control de Procesos Industriales y Robótica y Visión. En el curso 04/05 ha sido la intensificación mayoritariamente elegida por los alumnos de Electrónica de la EUITI-UPM. Actualmente hay alrededor de 25 alumnos matriculados en esta intensificación.

Actualmente el temario se encuentra en estado de estudio. No obstante, hay dos pilares fundamentales: los sistemas en tiempo real y la programación distribuida. El primero está puesto por que más del 90% de los microprocesadores fabricados en el mundo están dedicados a los sistemas embebidos[7]. El segundo hace referencia a una tendencia actual de que muchos algoritmos desarrollados para la industria requieren de una alta carga computacional y su resolución requiere un procesamiento distribuido, como una solución de bajo costo. La tabla adjunta las líneas principales de actuación de esta asignatura.

Tabla 5 Líneas de Ingeniería de la Programación

Temas	Contenido
Introducción a los sistemas en tiempo real	Características. Especificaciones. Planificación.
Fiabilidad y tolerancia a fallos	SW redundante. Medida y predicción de la fiabilidad.
Excepciones y manejo de las excepciones	Construcción de sistemas estables. Manejo de excepciones en C++. Programación defensiva.
Programación concurrente	Concurrencia en C++. Usando hebras. Compartiendo recursos. Terminando tareas. Cooperación entre hebras.
Sistemas distribuidos	Elementos de diseño en los SO. Invocación de servicios remotos. Archivos distribuidos.

En cuanto a la biografía se busca una conexión entre el curso de tecnología orientada a objetos, dada en Informática Industrial, con los sistemas en tiempo real y la programación distribuida. Se empleará el C++ como lenguaje en la construcción de estos sistemas, por eso, se apuesta por libros como:

- Bruce Eckel, Chuck Allison, *Thinking In C++ Volume 2: Practical Programming*, Utah Valley State College, www.bruceckel.com

- Cameron Hughes and Tracey Hughes, *Parallel and Distributed Programming Using C++*, Addison Wesley, 2004.

Referencias

- [1] Ceballos, F., (1995) *C/C++ Curso de programación*. RaMa.
- [2] www.abet.org
- [3] Larman, C., (2002) *Applying UML and Patterns*. Prentice Hall.
- [4] Gamma, E., Helm, R., Johnson, R., Vissides, J., (1995) *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.
- [5] Cockburn, A.(2001), *Writing Effective Use Cases*, Addison-Wesley.
- [6] Beck, K., (2000) *Extreme Programming Explained*, Addison Wesley.
- [7] Burn, A., Wellings, A., (2002) *Sistemas de tiempo real y Lenguajes de Programación*, Addison Wesley, 3ª edición.