

Carlos Gregorio Rodríguez<sup>1</sup>, Ángel Herranz Nieva<sup>2</sup>, Raquel Martínez Unanue<sup>3</sup>

<sup>1</sup> Dpto. Sistemas Informáticos y Programación, Facultad de Informática, Universidad Complutense de Madrid; <sup>2</sup> Dpto. Lenguajes y Sistemas, Facultad de Informática, Universidad Politécnica de Madrid; <sup>3</sup> Dpto. Ciencias Experimentales e Ingeniería, Universidad Rey Juan Carlos (Madrid)

<cgregor@slp.ucm.es>  
<aherranz@fi.upm.es>  
<r.martínez@escet.urjc.es>

**Resumen:** en diciembre de 2001 se publicó el informe final del volumen *Computer Science del Computing Curricula 2001*. Las recomendaciones desarrolladas por ACM y IEEE de forma individual, tanto como las recomendaciones elaboradas de forma conjunta, han sido y siguen siendo un punto de referencia para todos los que nos dedicamos a la enseñanza universitaria de la Informática. El objetivo de este artículo es dar a conocer las líneas maestras del informe, los principios del mismo, las nuevas áreas de conocimientos y las diferentes estrategias de implementación sugeridas. El objetivo final es que un lector interesado pueda hacerse una idea clara de las propuestas fundamentales del informe.

**Palabras clave:** planes de estudio, enseñanza de la Informática, recomendaciones curriculares, modelos curriculares, enseñanza universitaria, Informática.

## 1. Introducción

En 1998, la *Association for Computing Machinery (ACM)* y la *Computer Society of the Institute for Electrical Engineers (IEEE-CS)* volvieron a aunar esfuerzos y formaron un equipo de trabajo denominado *The Joint Task Force on Computing Curricula IEEE-CS/ACM* encargado de elaborar un informe con el siguiente propósito: «Revisar el currículum del año 1991 [2] (CC1991) para desarrollar una versión nueva y más elaborada del mismo en el año 2001 en la que se tuvieran en cuenta los avances tecnológicos acontecidos en la última década y que se desarrollarán en la próxima».

El fruto de dicho trabajo es el primero de los volúmenes de los nuevos currícula cuya versión final fue publicada el día 15 de diciembre de 2001 bajo el título: *Computing Curricula 2001, Computer Science Volume* [5]. Este artículo es el resultado de un estudio de dicho volumen.

Antes de entrar en materia conviene hablar brevemente de la estructura del informe completo: *Computing Curricula 2001 (CC2001)* [3]. Está compuesto por cinco volúmenes: un volumen que ofrece una perspectiva general y otros cuatro dedicados cada uno a una determinada disciplina. El equipo inicialmente encargado de elaborar el currículum estaba compuesto por expertos en Informática (*Computer Science*) y no tardaron en observar dificultades para abarcar todo el ámbito relacionado. Su solución fue adoptar un enfoque más amplio y elaborar recomendaciones curriculares específicas para cada disciplina: Informática (*Computer*

## Computing Curricula 2001

Esta versión incorpora correcciones a la versión impresa de este mismo artículo, especialmente a la figura 2, con el exclusivo propósito de incorporarla a las páginas web de Novática en el sitio web de ATI

*Science*), Arquitectura de Computadores (*Computer Engineering*), Ingeniería del Software (*Software Engineering*) y Sistemas de Información (*Information Systems*). Dicha estructura supone un cambio sustancial con respecto al informe CC1991 que constaba de un único volumen.

El volumen que no está dedicado a ninguna disciplina (*Overview*) ofrece una perspectiva general de todos los currícula y responde a dos necesidades básicas: (1) Definir el ámbito de las distintas disciplinas y el conjunto de temas común puesto que, tal y como reconoce el propio equipo, «*resulta complicado tanto a estudiantes como a docentes encontrar definiciones precisas y concisas de los distintos campos de la Informática*». (2) Definir principios, estrategias y guías para crear y mantener todos los volúmenes. Sólo el volumen dedicado a Informática [5] está terminado. Con respecto al resto, sólo el de Ingeniería del Software [6] y el de Sistemas de Información [4] se encuentran en fase de desarrollo. El resto del artículo se centra en el volumen de Informática y nos referiremos al mismo como *el informe CC2001* o simplemente como *el informe*.

Para poder hacer un estudio del currículum en Informática con una perspectiva más amplia que aquella contemplada en el informe CC1991, el equipo de trabajo se encargó de crear dos tipos de grupos: aquéllos encargados de identificar y organizar los contenidos de la disciplina en áreas de conocimiento (*knowledge focus groups*) y aquellos que tenían como objetivo el estudio de aspectos pedagógicos que dieran al informe definitivo una consistencia mayor que la de anteriores intentos (*pedagogy focus groups*).

El propósito de este artículo es dar a conocer las líneas maestras de dicho documento, sus principios, las nuevas áreas de conocimiento consideradas y las estrategias de implementación propuestas. Hemos pretendido ofrecer una visión general que facilite a los interesados la lectura y el entendimiento del mismo así como ofrecer una guía para acceder a la información detallada que el lector pueda necesitar. Hacemos por tanto nuestras las palabras que recogen el objetivo principal del informe [5]: «*Este informe ha sido diseñado principalmente como un recurso para aquellas escuelas y universidades que pretendan desarrollar o mejorar los programas de primer grado en Informática.*»

Los destinatarios de este artículo son fundamentalmente los profesores que imparten docencia en las titulaciones de Informática, así como los organismos involucrados en la

elaboración de los currícula. Los autores somos también profesores y hemos intentado extraer del informe aquellos aspectos que nos parecieron más relevantes de cara a entenderlo y poder valorar su posible aplicación en las universidades españolas.

La organización del artículo es la siguiente: en la sección 2 se describen los principios esenciales del informe; en la sección 3 el lector puede encontrar la identificación y organización de los contenidos del currículum para Informática realizada por el equipo de trabajo; en la sección 4 se ofrece una guía para que las instituciones puedan aplicar y adaptar el informe; en la sección 5 se describen algunas directrices para completar el currículum; en la sección 6 puede encontrarse información sobre un conjunto deseable de habilidades en los titulados en Informática; en la sección 7 nos referimos a algunas recomendaciones del informe con respecto al diseño de asignaturas de Informática para no informáticos. El artículo finaliza con la sección 8 en la que se recogen las conclusiones de los autores tras el estudio del informe.

En la **figura 1** se ha recogido el índice del informe. A lo largo del presente artículo remitiremos al lector a la parte, o partes, del informe en la que puede encontrar más detalles sobre los puntos comentados.

1.	Introducción	1
2.	Lecciones de anteriores informes	6
3.	Cambios en la disciplina Informática	9
4.	Principios	12
5.	Visión general del cuerpo de conocimientos	14
6.	Visión general de los modelos curriculares	18
7.	Cursos introductorios	22
8.	Cursos intermedios	35
9.	Completando el vitae	40
10.	Prácticas profesionales	55
11.	Características de los licenciados en Informática	62
12.	Informática a través del vitae	67
13.	Desafíos institucionales	74
A.	El cuerpo de conocimientos	83
B.	Descripción de los cursos	157

Figura 1. Índice del informe

## 2. Desafíos y principios

Uno de los principales problemas detectados en la evaluación del informe CC1991 es que éste se articulaba entorno a la definición de un conjunto de áreas y unidades de conocimiento en las que se organizaban los contenidos, mientras que se ofrecían muy pocos detalles sobre la descripción de los posibles cursos, es decir, sobre las posibles implementaciones finales del informe. Para resolver este problema y ofrecer mayor apoyo en el diseño de los currícula, el informe CC2001 propone varias alternativas para la organización de las unidades de conocimiento en cursos coherentes. Trataremos en profundidad este tema en la sección 4.

Otro aspecto importante que se ha tenido en cuenta durante la elaboración del informe es la mayor relevancia que han cobrado en los últimos años temas como Internet y sus aplicaciones, tecnología de redes, gráficos y multimedia,

bases de datos relacionales, sistemas empotrados, interoperabilidad, programación orientada a objetos, uso de interfaces avanzadas de programación de aplicaciones, interacción hombre-máquina, seguridad y criptografía. Temas que se han incorporado a las áreas de conocimiento, en muchas ocasiones, en detrimento de otros que han ido perdiendo importancia en la última década.

El informe CC2001 se ha llevado a cabo teniendo en cuenta una serie de principios que aparecen en el capítulo 4 del informe; a continuación los agrupamos y resumimos:

**Principios relacionados con la propia naturaleza de la Informática.** Se reconoce a la Informática como un campo amplio que se extiende más allá de los límites de la ciencia de la computación, de ahí que el equipo de trabajo del informe CC2001 se haya planteado el desarrollo de distintas recomendaciones para cubrir otras disciplinas Informáticas. Asimismo, al extraer sus fundamentos de una amplia variedad de disciplinas, surge la necesidad de que los titulados sean capaces de utilizar conceptos de diversos campos. Por último, la rápida evolución de la Informática requiere una revisión continua del currículum, que debe ser sensible a los cambios tecnológicos, nuevos desarrollos pedagógicos y a la importancia del aprendizaje continuo.

**Principios relacionados con las unidades y cuerpo de conocimientos.** El informe CC2001 tiene que identificar los conocimientos y destrezas fundamentales que todo titulado debería poseer. También se plantean como principios que el cuerpo de conocimientos obligatorio sea lo más pequeño posible, y que se acompañe con una definición de cursos completos que sirvan de guía para el diseño de los currícula.

**Principios relacionados con el alcance y los aspectos que el informe debe recoger.** El informe pretende conseguir un alcance internacional y en su desarrollo se planteó contar con la participación de las instituciones educativas, del sector público y privado. Asimismo, el CC2001 se propuso plantear, no sólo recomendaciones curriculares de alto nivel, sino las estrategias y tácticas para su implementación, e incluir, además, la práctica profesional como una componente integral del currículum.

En el propio informe se reconoce que el seguimiento de estos principios ha sido desigual. Por ejemplo, el que se refiere al alcance internacional del CC2001 no se ha cumplido de forma satisfactoria, ya que el contenido del informe está profundamente influido por la estructura de los estudios de Informática en Estados Unidos. De la misma manera, la participación del sector privado en la elaboración del informe ha sido finalmente menor que la esperada.

## 3. Cuerpo de conocimientos

Una de las principales tareas necesarias para la definición del informe CC2001 es la identificación y organización de los contenidos adecuados para el currículum de una licenciatura en Informática: el cuerpo de conocimientos (*body of knowledge*); que aparece en el capítulo 5 del informe.

En las fases iniciales de elaboración del proyecto se identificaron las áreas en las que se dividía el cuerpo de conoci-

<p><b>D S. Matemática discreta (43)</b></p> <ul style="list-style-type: none"> <li>- <u>Funciones, relaciones y conjuntos</u> (6)</li> <li>- <u>Fundamentos de lógica</u> (10)</li> <li>- <u>Técnicas de demostración</u> (12)</li> <li>- <u>Técnicas básicas de conteo</u> (5)</li> <li>- <u>Grafos y árboles</u> (4)</li> <li>- <u>Probabilidad discreta</u> (6)</li> </ul> <p><b>PF. Fundamentos de programación (38)</b></p> <ul style="list-style-type: none"> <li>- <u>Construcciones fundamentales de programación</u> (9)</li> <li>- <u>Algoritmos y resolución de problemas</u> (6)</li> <li>- <u>Estructuras de datos fundamentales</u> (14)</li> <li>- <u>Recursión</u> (5)</li> <li>- <u>Programación dirigida por eventos</u> (4)</li> </ul> <p><b>AL. Algoritmos y complejidad (31)</b></p> <ul style="list-style-type: none"> <li>- <u>Análisis básico de algoritmos</u> (4)</li> <li>- Estrategias algorítmicas (6)</li> <li>- Algoritmos fundamentales de computación (12)</li> <li>- <u>Algoritmos distribuidos</u> (3)</li> <li>- Teoría básica de la computabilidad (6)</li> <li>- Las clases de complejidad P y NP</li> <li>- Teoría de autómatas</li> <li>- Análisis avanzado de algoritmos</li> <li>- Algoritmos criptográficos</li> <li>- Algoritmos geométricos</li> <li>- Algoritmos paralelos</li> </ul> <p><b>AR. Arquitectura y organización (36)</b></p> <ul style="list-style-type: none"> <li>- <u>Sistemas lógicos y sistemas digitales</u> (6)</li> <li>- Representación de los datos al nivel de la máquina (3)</li> <li>- Organización de la máquina al nivel de ensamblador (9)</li> <li>- <u>Organización y arquitectura del sistema de memoria</u> (5)</li> <li>- Interfaces y comunicación (3)</li> <li>- Organización funcional (7)</li> <li>- <u>Multiproceso y arquitecturas alternativas</u> (3)</li> <li>- Mejoras del rendimiento</li> <li>- Arquitectura para redes y sistemas distribuidos</li> </ul> <p><b>OS. Sistemas operativos (18)</b></p> <ul style="list-style-type: none"> <li>- <u>Visión general de los sistemas operativos</u> (2)</li> <li>- <u>Principios de los sistemas operativos</u> (2)</li> <li>- <u>Concurrencia</u> (6)</li> <li>- <u>Planificación y reparto</u> (3)</li> <li>- <u>Gestión de memoria</u> (5)</li> <li>- Gestión de dispositivos</li> <li>- Seguridad y protección</li> <li>- Sistemas de archivos</li> <li>- Sistemas de tiempo real y empujados</li> <li>- Tolerancia a fallos</li> <li>- Evaluación del rendimiento de sistemas</li> <li>- Lenguajes de <i>script</i></li> </ul> <p><b>NC. Informática orientada a la red (15)</b></p> <ul style="list-style-type: none"> <li>- <u>Introducción a la computación orientada a la red</u> (2)</li> <li>- <u>Comunicación y redes</u> (7)</li> <li>- Seguridad de redes (3)</li> <li>- <u>La web como ejemplo de computación cliente-servidor</u> (3)</li> <li>- Construcción de aplicaciones para la web</li> <li>- Gestión de redes</li> <li>- Compresión y descompresión</li> <li>- Tecnologías de datos multimedia</li> <li>- Computación inalámbrica y móvil</li> </ul> <p><b>PL. Lenguajes de programación (21)</b></p> <ul style="list-style-type: none"> <li>- <u>Visión general de los lenguajes de programación</u> (2)</li> <li>- Máquinas virtuales (1)</li> <li>- Introducción a la traducción de lenguajes (2)</li> <li>- Declaraciones y tipos (3)</li> <li>- Mecanismos de abstracción (3)</li> <li>- Programación orientada a objetos (10)</li> <li>- Programación funcional</li> <li>- Sistemas de traducción de lenguajes</li> <li>- Sistemas de tipos</li> <li>- Semántica de los lenguajes de programación</li> <li>- Diseño de los lenguajes de programación</li> </ul>	<p><b>HC. Interacción persona-ordenador (IPO) (8)</b></p> <ul style="list-style-type: none"> <li>- <u>Fundamentos de la interacción persona-ordenador</u> (6)</li> <li>- <u>Construcción de un interfaz gráfico simple</u> (2)</li> <li>- Evaluación de software centrada en la persona</li> <li>- Desarrollo de software centrado en la persona</li> <li>- Diseño de interfaces gráficas de usuario</li> <li>- Programación de interfaces gráficas de usuario</li> <li>- Aspectos de la IPO de los sistemas multimedia</li> <li>- Aspectos de la IPO de la colaboración y comunicación</li> </ul> <p><b>GV. Informática gráfica y visual (3)</b></p> <ul style="list-style-type: none"> <li>- <u>Técnicas fundamentales en gráficos</u> (2)</li> <li>- <u>Sistemas gráficos</u> (1)</li> <li>- Comunicación gráfica</li> <li>- Modelado geométrico</li> <li>- Representación básica</li> <li>- Representación avanzada</li> <li>- Técnicas avanzadas</li> <li>- Animación por computador</li> <li>- Visualización</li> <li>- Realidad virtual</li> <li>- Visión por computador</li> </ul> <p><b>IS. Sistemas inteligentes (10)</b></p> <ul style="list-style-type: none"> <li>- Aspectos fundamentales de los sistemas inteligentes (1)</li> <li>- Búsqueda y satisfacción de restricciones (5)</li> <li>- Representación del conocimiento y razonamiento (4)</li> <li>- Búsqueda avanzada</li> <li>- Representación del conocimiento y razon. avanzados</li> <li>- Agentes</li> <li>- Procesamiento del lenguaje natural</li> <li>- Aprendizaje automático y redes neuronales</li> <li>- Sistemas de planificación inteligentes</li> <li>- Robótica</li> </ul> <p><b>IM. Gestión de la información (10)</b></p> <ul style="list-style-type: none"> <li>- Modelos y sistemas de información (3)</li> <li>- Sistemas de bases de datos (3)</li> <li>- <u>Modelado de datos</u> (4)</li> <li>- Bases de datos relacionales</li> <li>- Lenguajes de consulta de bases de datos</li> <li>- Diseño de bases de datos relacionales</li> <li>- Procesamiento de transacciones</li> <li>- Bases de datos distribuidas</li> <li>- Diseño físico de bases de datos</li> <li>- Minería de datos</li> <li>- Almacenamiento y recuperación de datos</li> <li>- Hipertexto e hipermedia</li> <li>- Información y sistemas multimedia</li> <li>- Bibliotecas digitales</li> </ul> <p><b>SP. Aspectos sociales y profesionales (16)</b></p> <ul style="list-style-type: none"> <li>- <u>Historia de la Informática</u> (1)</li> <li>- <u>Contexto social de la Informática</u> (3)</li> <li>- <u>Métodos y herramientas de análisis</u> (2)</li> <li>- <u>Responsabilidades éticas y profesionales</u> (3)</li> <li>- <u>Riesgos y responsabilidades de los sist. informáticos</u> (2)</li> <li>- <u>Propiedad intelectual</u> (3)</li> <li>- <u>Privacidad y libertades civiles</u> (2)</li> <li>- Crimen informático</li> <li>- Aspectos económicos de la Informática</li> <li>- Marcos filosóficos</li> </ul> <p><b>SE. Ingeniería del software (31)</b></p> <ul style="list-style-type: none"> <li>- Diseño de software (8)</li> <li>- Utilización de APIs (5)</li> <li>- <u>Herramientas y entornos del software</u> (3)</li> <li>- Procesos del software (2)</li> <li>- Requisitos y especificaciones software (4)</li> <li>- Validación del software (3)</li> <li>- Evolución del software (3)</li> <li>- Gestión de proyectos del software (3)</li> <li>- Software basado en componentes</li> <li>- Métodos formales</li> <li>- Fiabilidad del software</li> <li>- Desarrollo de sistemas especializados</li> </ul> <p><b>CN. Ciencia computacional</b></p> <ul style="list-style-type: none"> <li>- Análisis numérico</li> <li>- Investigación operativa</li> <li>- Modelos y simulación</li> <li>- Computación de altas prestaciones</li> </ul>
--	--

Figura 2. Cuerpo de conocimientos (traducción de la figura 5-1 del informe CC2001)

mientos. Frente a las nueve áreas que aparecían en el informe CC1991, el nuevo informe propone catorce áreas. Para cada una de estas áreas se designó un grupo de trabajo (*knowledge focus group*) compuesto por expertos y profesores con experiencia en la misma que definió y organizó sus contenidos. Cada área se divide en unidades (*units*) que son módulos temáticos individuales. Las unidades, a su vez, se dividen en temas (*topics*). En la **figura 2** se encuentran las catorce áreas y sus unidades.

Aquellas unidades de conocimiento que son consideradas por amplio consenso como esenciales para cualquier licenciatura en Informática han sido calificadas como *troncales* (*core* en el informe). El resto de las unidades son *no troncales* (*elective* en el informe). El conjunto de unidades troncales se denomina el *core*<sup>1</sup> en el informe. En la **figura 2** las unidades troncales aparecen subrayadas.

Consciente de la controversia o malentendidos que este tipo de división de las unidades de conocimiento puede ocasionar, el grupo de trabajo se encarga de matizar en detalle qué se entiende por unidad troncal:

- Las unidades troncales son aquéllas que se encuentran en todos los programas de licenciatura en Informática. El no ser troncal no significa tener menos importancia, sino que no hay un consenso amplio para su inclusión en el currículum.
- Las unidades troncales son imprescindibles pero no dan lugar a un currículum completo. Es necesario completarlo con material adicional.
- Las unidades troncales no son necesariamente aquéllas que pertenecen a cursos introductorios. El adjetivo troncal define la necesidad pero no el nivel.

Para dar una idea del tiempo que se necesita para impartir una unidad de conocimiento, se ha utilizado el número de horas de clase (lección magistral) que se necesitan para presentar dicho material. Ésta es la misma medida que han utilizado informes anteriores. En la **figura 2**, al lado de cada unidad, aparece el número mínimo de horas requerido para impartirla. Nuevamente, el informe pone cierto énfasis en matizar esta elección:

- No recomienda únicamente la lección magistral. El tiempo asignado a cada unidad debe considerarse como medida comparativa independientemente del estilo de enseñanza.
- Las horas sólo indican el tiempo de clase. Como norma general, se considera tres veces mayor el tiempo de trabajo requerido fuera de clase.
- Las horas que aparecen en la **figura 2** son las mínimas imprescindibles.

Una descripción detallada del cuerpo de conocimientos se encuentra en el apéndice A del informe, en el que se describen las áreas y para cada una de las unidades los objetivos de aprendizaje, y se desglosan los temas que la conforman.

Destacamos a continuación los cambios más sustanciales en cuanto a las áreas de conocimiento se refiere que presenta el informe CC2001 con respecto al anterior de 1991:

- «Matemática discreta» pasa de ser un prerrequisito para algunos contenidos de Informática en el informe CC1991 a un área temática componente del currículum.
- «Fundamentos de programación» pierde el carácter optativo que tenía en el CC1991 y pasa a ser una de las áreas de conocimiento tras la constatación de que muchos alumnos llegan al nivel universitario sin conocimientos sólidos en los aspectos básicos de la programación.
- Aparecen como nuevas áreas de conocimiento «Computación orientada a la red» e «Informática gráfica y visual» en respuesta a la relevancia que han cobrado estos temas en los últimos años.
- «Aspectos sociales y profesionales» pasan de ser unos contenidos especialmente recomendables en el informe CC1991 a un área del cuerpo de conocimientos en el informe CC2001. En los capítulos 9 y 11 del informe, a la hora de describir las destrezas que un titulado debe tener, se hace especial hincapié en aspectos como trabajo en equipo, facilidades para la comunicación, responsabilidad profesional o necesidad de aprendizaje a lo largo de toda la vida.

#### 4. ¿Cómo construir un currículum?

Resulta de suma importancia advertir de nuevo que el cuerpo de conocimientos descrito en el informe CC2001 no es un currículum en sí mismo. Como ya hemos señalado, posiblemente una de las principales desventajas que presentaba el informe CC1991 fuera la carencia de modelos estándares de currícula que facilitarían su adopción por parte de las escuelas y facultades de Informática. En el informe CC2001 se ha pretendido solventar este problema mediante una definición precisa de estrategias para construir un currículum (capítulos 6 al 9 del informe). La filosofía se basa en la división del currículum en tres fases: introductoria, intermedia y avanzada; y en la descripción de distintas estrategias para las dos primeras fases: seis para la introductoria y cuatro para la intermedia, tal y como se recoge en la **figura 3**.

<b>Cursos introductorios</b>	Primero Imperativa	Primero Objetos	Primero Funcional	Primero Global	Primero Algoritmos	Primero Hardware
<b>Cursos intermedios</b>	Enfoque tradicional basado en temas	Enfoque concentrado	Enfoque basado en sistemas	Enfoque basado en web		
<b>Cursos avanzados</b>	Cursos adicionales para complementar el currículum					

**Figura 3.** Estrategias de implementación (traducción de la figura 6-1 del informe CC2001)

Si una institución pretende adherirse al currículum CC2001 para Informática, deberá comenzar por elegir una implementación de la fase introductoria y otra de la fase intermedia y deberá asegurar que su programa cubre todas las asignaturas troncales del cuerpo de conocimientos. Aun así, esto no es suficiente, puesto que dicha institución deberá garantizar que los alumnos reciban una formación adecuada en determinadas áreas a través de las asignaturas de la fase avanzada. Además, la implementación específica deberá tener en cuenta las características propias de la institución, las preferencias de su profesorado y las necesidades de sus estudiantes. A lo largo del informe se pueden encontrar los detalles precisos sobre cómo proceder y en el apéndice B del mismo aparecen varios ejemplos con la descripción de diversas alternativas de implementación, incluyendo las asignaturas y sus contenidos.

La principal ventaja de esta estructura es que ofrece una gran flexibilidad. Las diferentes estrategias de la fase introductoria deben implementarse de forma que se alcancen una serie de objetivos comunes, recogidos en las figuras 7-1 y 7-2 del informe, lo que permite implementar cualquier estrategia intermedia mientras se tenga en cuenta ese conocimiento común. Además, con ello se permite que una institución opte por una estrategia en una de las fases y posteriormente la cambie por otra puesto que el impacto provocado por dicho cambio queda minimizado.

El resto de esta sección está dedicado a recoger los puntos que hemos considerado claves para entender tanto el proceso de adopción de las directivas del informe CC2001 como su justificación. Comenzaremos por describir las ideas que hay detrás de los currícula introductorio e intermedio, para terminar con algunas sugerencias para el currículum avanzado.

#### 4.1. El currículum introductorio

En esta fase del currículum se englobarían las asignaturas de primer y segundo año, es decir, las que serán la toma de contacto del alumno con la Informática. El informe no defiende un modelo curricular concreto: “[...] *lo cierto es que todavía no se ha encontrado una estrategia ideal para comenzar a enseñar Informática [...]*”. Por ello, se ofrecen seis estrategias de implementación para la fase introductoria como pueden verse en la **figura 3**. Pasamos a describir brevemente cada una de ellas.

**Primero imperativa.** Es sin duda la estrategia más tradicional de todas, con las ventajas e inconvenientes que ello conlleva. Exponemos a continuación una de las implementaciones propuestas en el informe<sup>2</sup>:

**CS101I.** Fundamentos de programación.

**CS102I.** El paradigma de orientación a objetos.

**CS103I.** Estructuras de datos y algoritmos.

**Primero objetos.** Un enfoque también centrado en la programación pero con especial atención a la programación y el diseño orientado a objetos desde el principio. Veamos una de las propuestas en el informe:

**CS101O.** Introducción a la programación orientada a objetos.

**CS102O.** Orientación a objetos y abstracciones de datos.

**CS103O.** Estructuras de datos y algoritmos.

**Primero funcional.** Sus principales ventajas son que resulta más sencillo centrarse en temas fundamentales y que permite que conceptos de suma importancia aparezcan de forma natural. Su desventaja: los alumnos no reaccionan positivamente al encontrar un lenguaje «extraño». Una propuesta de implementación:

**CS111F.** Introducción a la programación funcional.

**CS112F.** Orientación a objetos y algoritmos.

**Primero una visión global.** Esta estrategia nace como alternativa al resto de estrategias, por ser éstas excesivamente orientadas a la programación. Aunque su justificación en el informe es algo oscura, en la siguiente implementación se puede intuir la idea:

**CS101B.** Introducción a la Informática.

**CS102B.** Algoritmos y técnicas de programación.

**CS103B.** Diseño orientado a objetos.

**Primero algoritmos.** En esta estrategia se apuesta por no usar un lenguaje de programación sino un pseudocódigo para mostrar los conceptos básicos. Una propuesta de asignaturas:

**CS111A.** Introducción a los algoritmos y aplicaciones.

**CS112A.** Metodología de la programación.

En el informe se sugiere que esta estrategia esté sincronizada con trabajos prácticos.

**Primero hardware.** En esta estrategia se opta por enseñar el hardware primero para construir sobre dicho conocimiento los conceptos abstractos necesarios. Una de las propuestas es la siguiente:

**CS111H.** Introducción a los computadores.

**CS112H.** Técnicas de programación orientada a objetos.

Una información mucho más detallada de cada una de estas estrategias puede encontrarse en el capítulo 7 del informe CC2001. Junto a ésta aparece la respuesta a una pregunta clave para elaborar estrategias para el currículum introductorio: *¿Cuáles son los puntos que pueden considerarse esenciales en el diseño de la primera fase del currículum?* El informe CC2001 considera tres puntos centrales y discute sobre ellos con bastante profundidad: el papel de la programación, la duración de la fase introductoria y la integración de la matemática discreta; para ofrecer finalmente una tabla relativamente exhaustiva de los conceptos que debe cubrir esta fase (figura 7-1 y sección 7.5 del informe). Consideramos la lectura completa y cuidadosa de dicho capítulo como de especial relevancia para cualquier lector interesado en el diseño de un currículum.

#### 4.2. El currículum intermedio

Las asignaturas en esta fase del currículum suelen ofrecerse en segundo y tercer año, cuarto incluso si hablamos de algunas universidades europeas y sin duda españolas. Estas asignaturas son las encargadas de construir una buena base de conocimientos para que el alumno sea capaz de realizar estudios más avanzados; de hecho, es la fase en la que se concentran la mayor parte de las asignaturas troncales.

Al comienzo del capítulo 8 del informe CC2001 se recoge una cita de [8] con la que el grupo de trabajo pretende transmitir la idea de que es necesario realizar esfuerzos para

que los currícula se elaboren en torno a los conceptos abstractos que fundamentan la Informática, abandonando un legado histórico de dudosa utilidad actualmente: *«Organicemos nuestras asignaturas en torno a las ideas en lugar de hacerlo en torno a los productos. Esto ayuda a que los objetivos queden claros tanto para los estudiantes como para los docentes. Las escuelas de ingeniería no enseñan diseño de máquinas de vapor, enseñan termodinámica. Todavía pueden encontrarse dos asignaturas, construcción de compiladores y sistemas operativos, que son dinosaurios informáticos».*

Desde nuestro punto de vista, es obvio lo complicado de evitar una fase intermedia basada en los productos; sin embargo, no debería resultar tan complejo revisar si los productos que en su momento fueron clave (como los compiladores o los sistemas operativos) siguen siéndolo actualmente y si todavía es admisible que estas asignaturas desplacen a otras como por ejemplo las relacionadas con redes de computadores o componentes software.

Para la fase intermedia, se describen en el informe cuatro estrategias de implementación que no pretenden ser prescriptivas puesto que admiten implementaciones híbridas de las mismas:

- **Enfoque tradicional basado en temas.** Consiste en la estrategia más habitual; en ella, las asignaturas surgen de una división tradicional de las áreas de la Informática: análisis y diseño de algoritmos, arquitectura de computadoras, sistemas operativos, bases de datos, etc.
- **Enfoque concentrado.** Esta estrategia está especialmente pensada para aquellas instituciones que necesitan reducir el número de cursos de la fase intermedia y la idea consiste en concentrar distintos conocimientos de áreas relacionadas en una misma asignatura.
- **Enfoque basado en sistemas.** Este enfoque utiliza la idea de desarrollo de sistemas aplicada a las diversas áreas con el objetivo de tratarlas de forma unificada desde ese punto de vista. Con este fin se incluye más material técnico y profesional que en los otros modelos.
- **Enfoque basado en web.** En este caso, la implementación se centra en la preparación de los alumnos en tecnología relacionada con Internet y sus aplicaciones. En el informe se justifica su inclusión dada la creciente demanda en dicho campo.

#### 4.3. El currículum avanzado

El informe resalta en varias ocasiones que los grupos de asignaturas de la fase introductoria y de la fase intermedia no pueden en ningún caso constituir un currículum completo. Es necesario que los programas de primer grado incluyan una cantidad importante de material que asegure que los estudiantes tienen una formación y unas capacidades que necesitarán con posterioridad, así como la oportunidad de realizar trabajos avanzados que vayan mucho más allá del conjunto de unidades de conocimiento troncales.

En la sección 9.1 del informe el lector puede encontrar una descripción detallada de los requisitos esenciales para cerrar un currículum: rigor matemático, el método científico, familiarización con las aplicaciones, capacidad de comunicación y trabajo en equipo; sobre ellos abundaremos en la

sección 6. Las asignaturas avanzadas suelen ofrecerse en los dos últimos años. Para la implementación de la esta fase, el informe ofrece una lista de asignaturas por área así como una serie de modelos dependiendo del estilo de la universidad (figuras 9-1 y 9-2 en el informe).

## 5. Completando el currículum

En esta sección se presentan dos necesidades señaladas por el informe CC2001: la incorporación de prácticas profesionales al currículum y la adaptación del informe a las necesidades tácticas y estratégicas de las instituciones.

### 5.1. Prácticas profesionales

En la elaboración del informe CC2001 se ha tratado de recoger la creciente demanda y necesidad social de profesionales en Informática. En particular, el capítulo 10 se dedica íntegramente a las prácticas profesionales.

El informe afirma que la demanda de profesionales preparados para incorporarse rápidamente a sus puestos de trabajo hace que las prácticas profesionales sean cada vez más importantes y necesarias. Las empresas buscan estudiantes con una serie de habilidades (comunicación verbal y escrita, trabajo en equipo, motivación, etc.) y por tanto, el desarrollo de tales habilidades debe formar parte del currículum.

Se presta especial atención a la preparación y formación de los alumnos en asuntos de ética y códigos de conducta, recomendando que los profesores formen parte de asociaciones de profesionales y que animen a los estudiantes a participar en ellas.

El informe describe los mecanismos habituales para incorporar las prácticas profesionales dentro del currículum (proyecto final de carrera, prácticas en empresas...) pero plantea, además, la inclusión de las mismas a lo largo del currículum de modo transversal, dando ideas para cada uno de los niveles. En cursos básicos, con discusiones o trabajos sobre el impacto de la Informática en la sociedad o sobre la importancia de la práctica profesional, etc. En cursos intermedios, con historia de la Informática, problemas éticos de la vida real, estándares, etc. y, siempre que sea posible, desarrollando proyectos para clientes externos. En cursos avanzados, con cursos especialmente dedicados a ética o a leyes Informáticas o el trabajo de final de carrera.

Otro mecanismo que se propone es el de la valoración profesional del trabajo práctico. Se aconseja la creación de infraestructuras para que el trabajo de los alumnos sea valorado bajo estándares comunes y donde se fomente un acabado profesional.

En el informe también se analiza la forma en que las prácticas profesionales pueden ser potenciadas y financiadas desde tres niveles distintos: la industria y la administración, los departamentos y la universidad.

### 5.2. Adaptando el informe

En el informe CC2001 se reconoce la necesidad de adaptar los detalles del mismo, especialmente aquéllos que pueden

encontrarse en sus apéndices, a las necesidades estratégicas y tácticas de las instituciones que intenten adoptarlo. En el capítulo 13 del informe CC2001 pueden encontrarse innumerables factores que deberán tenerse en cuenta a la hora de aplicarlo, como el tipo de institución, la preparación de los alumnos que acceden, el interés y el grado de preparación de los profesores, las limitaciones de infraestructura, los costes, etc.

Nos gustaría resaltar un punto de especial interés: el profesorado. En el informe se trata la dificultad de atraer y retener a un profesorado cualificado ante la situación actual del mercado de trabajo en Informática. Se proponen varias estrategias para obtener dicho objetivo y se ofrecen algunos ejemplos: un plan agresivo de captación de profesorado, la creación de puestos específicos para la enseñanza, asegurar los recursos del profesorado e involucrar a los alumnos en labores de enseñanza.

## 6. Habilidades de un titulado

El objetivo último de unas recomendaciones curriculares es que los titulados adquieran una serie de conocimientos y habilidades. Dichos conocimientos y habilidades se describen en el capítulo 11 del informe, aunque también hay una relación en la sección 9.1. Todos estos contenidos están basados en un informe desarrollado para los titulados en Informática en Gran Bretaña [1].

Un resumen de las características generales que los titulados en Informática deberían poseer es el siguiente:

- Perspectiva global de sistemas que trascienda a los detalles de implementación de los diferentes componentes de un sistema informático.
- Apreciación de la interacción entre teoría y práctica.
- Familiaridad con los temas comunes (abstracción, complejidad, cambio evolutivo), de forma que sepan aplicarlos a dominios diferentes de aquellos en los que se aprendieron.
- Experiencia significativa en al menos un proyecto de software.
- Adaptabilidad para mantener las destrezas y habilidades a medida que la Informática evoluciona.

En la figura 11-1 del informe se detallan las aptitudes, destrezas y habilidades de un titulado en Informática divididas en tres categorías: cognitivas, prácticas y adicionales. Por otra parte, un titulado en Informática debe tener ciertas habilidades y destrezas más allá del cuerpo de conocimientos:

- Se reconoce la importancia y necesidad de que los titulados adquieran un cierto nivel de sofisticación matemática. Con este fin, se propone la introducción de técnicas matemáticas y métodos formales en los cursos introductorios para que puedan ser utilizados regularmente en cursos posteriores.
- Con el objetivo de que los titulados puedan «hacer ciencia y no sólo leer ciencia» el informe CC2001 recomienda que tengan una experiencia directa del método científico; es decir, formulación de hipótesis, diseño de la parte experimental, evaluación de hipótesis y análisis de los datos.
- Hay numerosas áreas de aplicación de la Informática, de ahí, que resulte útil para los titulados abordar el estudio de un área ajena que utilice la Informática. Con ello se

persigue que los titulados sean capaces de trabajar de forma satisfactoria con gente y en áreas de otras disciplinas.

- Los titulados deberían ser capaces de comunicar, de manera eficaz, ideas en forma escrita y oral, así como de valorar las presentaciones ajenas.
- Los proyectos en los que se verán involucrados la mayoría de los titulados se llevarán a cabo en grupos y, por tanto, deberán conocer los mecanismos y la dinámica del trabajo en equipo.
- Hay otras destrezas que en el informe se agrupan bajo el nombre «currículo complementario» que también son deseables: la capacidad de escribir resúmenes, realizar búsquedas bibliográficas, administrar el tiempo de forma eficiente, mantener la responsabilidad profesional, aprendizaje continuo, etc.

## 7. Informática para no informáticos

En el capítulo 12 del informe se identifican aquellos aspectos de la Informática relevantes para otras disciplinas académicas y propone guías para ayudar a que los estudiantes no informáticos consigan dicho conocimiento. Todas las propuestas de este capítulo se basan en el informe [7].

La Informática no es sólo un área de estudio sino una herramienta fundamental para muchas otras disciplinas. Los departamentos de Informática deben ser conscientes de ello y hacerse responsables de ofrecer una educación de alta calidad para una gran variedad de estudiantes. Por su parte, las universidades, dado el impacto de la Informática en muchos aspectos de la sociedad, tienen la responsabilidad de ofrecer cursos en Informática a todos los estudiantes y de garantizar que los departamentos tengan los recursos necesarios.

El mismo capítulo propone un repaso al método de definición de un curso teniendo en cuenta que los alumnos destinatarios pueden ser no informáticos. Las fases de desarrollo de un curso propuestas en el informe son especificación, diseño, implementación y valoración. En cada una de éstas se propone plantear de nuevo algunas cuestiones que pueden ser muy distintas a la hora de afrontar la preparación de un curso para no informáticos. Un breve resumen de cada fase es:

- **Especificación.** Los objetivos del curso deben ser discutidos por informáticos y no informáticos.
- **Diseño.** Los objetivos educativos tienen que ser concretados. Las destrezas y conceptos que deben alcanzar los alumnos deben ser claramente especificadas.
- **Implementación.** La estructuración del curso debe tener en cuenta los conocimientos previos, los métodos de trabajo, las motivaciones, los recursos, etc. de los alumnos a los que va dirigido el curso.
- **Valoración.** La valoración de los cursos impartidos debe ser activa y debe redundar en una constante actualización.

Además, el informe propone tres modelos diferentes de curso para no informáticos.

- **Cursos de introducción general.** Destinados a satisfacer el interés en Informática sin desarrollar ninguna habilidad en particular.
- **Cursos destinados a un área.** Cursos que son válidos para

estudiantes de ciertas titulaciones, por ejemplo inteligencia artificial para ciencias cognitivas, o Informática gráfica para diseño y bellas artes, o modelos informáticos para las titulaciones de economía y empresa.

- **Cursos especializados.** Son aquellos que necesitan un grupo de alumnos muy concreto dentro de una titulación o especialización. Por ejemplo, biología computacional, o física computacional.

En el informe [7] se incluyen descripciones detalladas de los cursos anteriormente descritos. La mayoría de los estudiantes universitarios va a recibir, a lo sumo, un único curso en Informática a lo largo de su carrera por lo que es muy importante que dichos cursos estén diseñados cuidadosamente para hacerlos tan útiles como sea posible.

## 8. Conclusiones

A pesar de las controversias que cualquier informe de este tipo puede levantar, consideramos que el CC2001 es el informe curricular más elaborado y exhaustivo de cuantos ha habido hasta la fecha. El informe ha sido fruto del esfuerzo de una gran cantidad de personas relacionadas con la Informática y demuestra la madurez tanto docente como científica que va alcanzando nuestra disciplina.

Para terminar, nos gustaría expresar algunas matizaciones al informe que podrían escaparse, o inducir a error, en una lectura poco detallada.

La descripción de las estrategias de implementación en algunos casos resulta un tanto oscura y su denominación puede llevar a confusión. Veamos algunos ejemplos. Cuando se describen las estrategias de implementación de la fase introductoria se indica que las tres primeras (primero imperativa, primero objetos y primero funcional) corresponden al modelo primero programación (*programming-first*) mientras que las otras tres (primero una visión global, primero algoritmos y primero *hardware*) adoptan un paradigma alternativo. Sin embargo, si se profundiza y se observan los cursos que proponen para cada estrategia, en todas se incluye la programación y, en particular, la programación orientada a objetos como parte sustancial. No podría ser de otra manera para garantizar que desde cualquier estrategia del primer nivel se pueda pasar a cualquier estrategia del segundo nivel.

Otro punto que nos ha llamado la atención es el enfoque basado en web del nivel intermedio. Se incluye, según indican, por la creciente demanda en dicho campo, pero no está claro que este punto de vista de la disciplina esté suficientemente cristalizado como para dar lugar a un enfoque realmente diferente y robusto. Por otra parte, puede entrar en contradicción con su crítica al diseño de una fase basada en los productos.

En cuanto a aspectos del informe que contrastan con la universidad española, destacamos los que se refieren a las prácticas profesionales. En el informe se hace especial énfasis en la conveniencia de realizar prácticas profesionales y propone diversos mecanismos según los niveles. La importancia que se da a este aspecto del currículum en el informe es considerablemente mayor que la que se le da, en

general, en las titulaciones de las universidades españolas. En éstas, las prácticas profesionales quedan reducidas a asignaturas de prácticas en empresas (optativas o de libre elección) que tienen escaso éxito entre los alumnos. Parte del fracaso de estas asignaturas se puede deber a una no muy fluida relación entre los departamentos universitarios y las empresas, que hace que la oferta de prácticas no sea suficientemente atractiva para los alumnos. Los proyectos de fin de carrera suelen ser la única ocasión en la que los alumnos abordan un proyecto de cierta envergadura, que engloba diversos aspectos de la disciplina que normalmente han estudiado de forma aislada, y en el que se les exige un acabado de calidad.

## Referencias

- [1] **Quality Assurance Agency for Higher Education.** *A report on benchmark levels for computing*, 2000. Gloucester, England: Southgate House.
- [2] **ACM/IEEE-CS Joint Curriculum Task Force.** *Computing Curricula 1991*. <<http://www.acm.org/education/curr91/>>
- [3] **The Joint Task Force on Computing Curricula IEEE-CS/ACM.** *Computing Curricula 2001*. <<http://www.computer.org/education/cc2001/>>
- [4] **The Joint Task Force on Computing Curricula IEEE-CS/ACM.** *Computing Curricula 2001 - Information Systems Volume* (en desarrollo). <<http://www.acm.org/education/curricula.html>>
- [5] **The Joint Task Force on Computing Curricula IEEE-CS/ACM.** *Computing Curricula 2001 - Computer Science Volume*, December 2001. <<http://www.computer.org/education/cc2001/final/>>
- [6] **The Joint Task Force on Computing Curricula IEEE-CS/ACM.** *Computing Curricula 2001 - Software Engineering Volume* (en desarrollo). <<http://sites.computer.org/ccse/>>
- [7] **Computing Science and Telecommunications Board.** *Being fluent with information Technology*. National Academy Press, 1999.
- [8] **Mary Shaw.** «We can teach software better». *Computing Research News*, 4(4):2-12, September 1992.

## Notas

<sup>1</sup> Obsérvese el uso de *core* como adjetivo y como sustantivo.

<sup>2</sup> La codificación asociada a cada curso está descrita en el apéndice B del informe. Los cursos son semestrales, de tres horas por semana, considerando un total de unas 40 horas lectivas. Lo que equivale a 4 créditos en nuestro sistema, a 3 créditos en los EE.UU. y a 15-16 puntos en el Reino Unido.