

Propuesta de contenido de prácticas en la materia de Sistemas Operativos

Miguel Riesco Albizu, Marián Díaz Fondón

Departamento de Informática

Universidad de Oviedo

Oviedo

e-mail: {albizu, fondon}@uniovi.es

Resumen

El contenido de la materia de Sistemas Operativos y su estructuración en forma de temario para una o varias asignaturas está bastante bien definido y aceptado. Sin embargo, la práctica unanimidad en el apartado de teoría se transforma en una gran variedad de enfoques, diametralmente opuestos en muchos casos, en lo que a las prácticas de la asignatura se refiere.

En este artículo se revisan distintas tendencias a la hora de enfocar la parte de prácticas de la asignatura, y se hace una propuesta de contenidos para ella.

1. Introducción

La materia de Sistemas Operativos es lo que podríamos denominar un “clásico” dentro de cualquier titulación informática. Los distintos modelos curriculares [35, 36], internacionalmente tomados como referencia, las directrices generales emanadas del Consejo de Universidades [22, 23, 24], así como la mayor parte de los libros de texto sobre la materia [19, 25, 31, 32, 34], coinciden en los contenidos temáticos que debe incluir la materia.

Dado que es potestad de cada Universidad y cada Centro articular las materias en asignaturas, la materia de Sistemas Operativos puede aparecer recogida en una o en varias asignaturas. Pero independientemente de cómo se estructure la materia en asignaturas el contenido conjunto y los objetivos de las mismas suelen ser similares: mostrar la finalidad, el funcionamiento y la estructura interna de los sistemas operativos como parte fundamental del sistema de computación.

Así las cosas, nos encontramos con una situación casi idílica para cualquier profesor que se enfrente a la docencia de una asignatura universitaria: un contenido bien establecido y no sujeto a discusión, poco variable en el tiempo

(como mucho recoger en el temario a modo de ejemplo las características de los nuevos sistemas que puedan aparecer, que no son muchos), una buena bibliografía tanto básica como de referencia, con un gran número de ejercicios y problemas sobre la materia, etc.

Pero esta situación ideal se transforma en incertidumbre cuando ese mismo profesor debe llevar a cabo el diseño del programa de las prácticas de la asignatura. A diferencia de otras asignaturas donde está claro el objetivo que se busca con las prácticas, en el caso de Sistemas Operativos hay distintas opciones a la hora de elegir el objetivo y, por consiguiente, diferentes tipos de prácticas que pueden llevarse a cabo.

Esta disparidad de tipos de prácticas queda patente cuando se hace un repaso de los temarios de las asignaturas que se imparten en los distintos Centros.

A continuación repasaremos los distintos enfoques que habitualmente se da a las prácticas de sistemas operativos, para, posteriormente, exponer nuestra propuesta.

2. Objetivos de las prácticas

Cuando revisamos el contenido de las prácticas de cualquier asignatura perteneciente a una carrera informática (y que podría ser aplicable a casi cualquier otro tipo de carrera) nos encontramos con distintos enfoques, dependiendo del tipo y de los objetivos de la asignatura de que se trate:

1. Un primer grupo estaría formado por aquellas asignaturas donde las prácticas son conducentes a ejercitar los conocimientos que se han explicado en la parte de teoría. Asignaturas como *Bases de Datos*, *Metodología de la Programación* o *Administración de S.S.OO.* son claros ejemplos de este tipo de prácticas, dado que en la parte de teoría se explican los conceptos que van a ser utilizados en la vida real en el ámbito correspondiente, y en este caso

las prácticas tienen el objeto de “simular”, de alguna manera, problemas relacionados con la materia de que se trate, con los que se puede encontrar el alumno en su vida laboral

2. Prácticas destinadas a reforzar lo expuesto en la parte de teoría. Este es el caso de asignaturas de tipo más básico, no tan finalista como en el caso anterior, donde los conocimientos que se imparten rara vez se aplicarán en la vida profesional del alumno, sino que son necesarios para asignaturas posteriores. *Arquitectura de computadores*, *Autómatas* o *Lógica* pueden encuadrarse en este grupo. Por ejemplo, en el caso de la primera asignatura citada una práctica típica consiste en diseñar algún tipo de procesador u otro elemento del computador o en la utilización del simulador de algún sistema (ficticio o real), con objeto de que el alumno comprenda mejor su funcionamiento básico.

La asignatura de Sistemas Operativos no encaja claramente en ninguno de los dos grupos. De hecho, la decisión de encuadrarla en el primer grupo, en el segundo, o incluso en ambos es un factor decisivo a la hora de explicar la causa de que existan muy distintos planteamientos en cuanto a su contenido y enfoque. Otros factores, como el número de créditos disponibles o el nivel de conocimiento previo de que dispongan los alumnos también condicionan el diseño final del programa de prácticas.

3. Tipos de prácticas existentes

Repasando los programas de la asignatura de Sistemas Operativos de distintos Centros y Universidades, podemos realizar una clasificación de los tipos de prácticas que se pueden llevar a cabo.

3.1. A nivel de usuario del sistema.

Consisten en el empleo de un Sistema Operativo como Unix, y la realización de prácticas sobre el intérprete de órdenes: órdenes básicas y programación del shell, proporcionando experiencia sobre la utilización al nivel de usuario del sistema operativo. Normalmente este tipo de prácticas se combinan con alguna otra de mayor nivel, como las prácticas a nivel de programador.

La Universidad de Oregón [9] realiza alguna de estas prácticas además de otros tipos. En

España, casi todas las universidades realizan una de estas prácticas, compaginada con alguna otra. La Universidad Sevilla [11] o la de Granada [3] son ejemplo de ello.

3.2. A nivel de programador del sistema

Empleo de llamadas al sistema dentro de programas. Fundamentalmente se utiliza el sistema operativo Unix, y el lenguaje C para realizar ejercicios que utilicen llamadas al sistema para gestión de procesos y gestión de ficheros. Este tipo de prácticas se abordan en la Universidad de Oregón entre otras. Dentro del territorio nacional, Politécnica de Madrid [16], Politécnica de Cataluña [15], incluyen alguna de estas prácticas.

Un modelo de práctica bastante utilizado es la implementación de un intérprete de órdenes para Unix, como ocurre en la Universidad de Brown [6] la Politécnica de Madrid o la Universidad de Granada.

Otro aspecto que suele trabajar son las llamadas de comunicación entre procesos o la gestión de hilos. Sevilla o Granada o la Politécnica de Cataluña son ejemplos de ello.

3.3. A nivel de diseñador del sistema

Se trata de realizar un diseño de una o varias partes del sistema operativo, aplicando los conceptos teóricos vistos en las clases. Dentro de esta modalidad, existen dos tendencias claramente diferenciadas:

3.3.1. Diseño a partir de un entorno para desarrollo de proyectos

Muchas Universidades a las que se ha tenido acceso a su información, utilizan herramientas especialmente diseñadas para su uso como parte de las prácticas de un curso de “Sistemas Operativos”. Tal es el caso del entorno Nachos [29], construido en la Universidad de Berkeley, y que constituye un esqueleto de sistema operativo simulado sobre una máquina abstracta, y que sirve de plataforma de desarrollo para que los alumnos implementen las funcionalidades que éste no posee.

Existe gran cantidad de documentación sobre este entorno, tanto para explicación del mismo

como para diseño de las prácticas que pueden plantearse sobre él. Algunas ediciones actuales de textos reconocidos como “Sistemas Operativos” de William Stallings [32], 3ª edición y posteriores, dedican un apéndice a esta herramienta. Los alumnos deberán aplicar los conocimientos de C++ y programación orientada a objetos para trabajar con Nachos.

Universidades como Chicago [7], o Duke [2], entre otras, utilizan Nachos como entorno base para el desarrollo de proyectos. En España, este entorno se utiliza con frecuencia en asignaturas de diseño de sistemas operativos, situadas en segundo ciclo puesto que el alumno necesita mayores conocimientos de programación. La propia Universidad de Oviedo, en su asignatura de segundo ciclo, dentro de la asignatura de Diseño de Sistemas Operativos [1], es un ejemplo de ello.

Existen otros entornos que también se usan como base para las prácticas. De características similares al Nachos está el OSP [26] (Entorno para “Operating System Projects”). Análogamente al Nachos, el OSP está basado en una máquina abstracta que simula el comportamiento del hardware. No es pues, un sistema operativo real. La Universidad Politécnica de Cataluña [15] emplea esta técnica, utilizando un núcleo de Sistema Operativo de desarrollo propio, denominado ONION, a partir del cual los alumnos añaden nuevas funciones. Harvard [8] utiliza también un núcleo propio para el apoyo al diseño de funcionalidad sobre él (OS/161).

Jesús Carretero, en su libro de prácticas de Sistemas Operativos [18], incluye un minikernel propio con la finalidad de servir como base para la ampliación de funcionalidad por parte de los alumnos. La Universidad Carlos III [5] lo usa para sus prácticas de Sistemas Operativos.

3.3.2. Diseño a partir de verdaderos Sistemas Operativos

Otro modo de abordar la implementación de las prácticas a partir de un entorno es mediante el uso de verdaderos Sistemas Operativos (en contraposición con los sistemas simulados como Nachos y OSP), se usan en este caso sistemas como Minix [28], Xinu [21], Linux [27], o Tunix [33]. El uso de este tipo de software introduce al estudiante en los detalles de más bajo nivel, que no están presentes en el simulador. La ventaja de

los simuladores es precisamente esa, centrar al estudiante en la implementación de los conceptos propios del sistema operativo discutidos en clase sin entrar en las particularidades de la máquina. La Universidad de Vrije [13] utiliza Minix.

3.3.3. Diseño de simulaciones de partes del sistema operativo partiendo de cero

Otra modalidad consiste en plantear un diseño de un hipotético sistema operativo simulado sobre la plataforma existente. En este caso se pueden omitir partes del sistema y centrarse sólo en la que en ese momento interesa. Es posible incluso plantear una serie de prácticas consecutivas en las que se van obteniendo diversos objetivos de forma que cada una es continuación de las anteriores, hasta conseguir una simulación considerable del sistema operativo.

Algunas universidades como PennState [10] o Wisconsin [14] utilizan este modelo de prácticas. Tiene la ventaja, frente al uso de un entorno predeterminado, de una mayor flexibilidad en cuanto a los niveles de dificultad que se pueden establecer, no necesitando una fase previa de aprendizaje del entorno de trabajo. El desarrollo de estas prácticas es también variado. Algunas universidades como Wisconsin incorporan ya el uso de Java como entorno de programación, mientras que otras trabajan con C o C++. En la Universidad Carlos III, Valladolid [12] y la E.U.I. Técnica en Informática de Gijón [4] también se usa este mecanismo en la realización de prácticas.

3.4. Módulo de programación concurrente

La programación concurrente, interbloqueos, semáforos, etc. que se sale de lo que es estrictamente el diseño de un sistema operativo, suele tener una práctica especialmente dedicada a ello.

Existen también herramientas que ayudan a la realización de estas prácticas. Actualmente se está utilizando la denominada BACI (The Ben-Ari Concurrent Programming System) [17], desarrollada por Bill Bynum, de la Universidad de Alabama. BACI simula la ejecución de procesos concurrentes y soporta semáforos generales, semáforos binarios, y monitores. Con anterioridad a este sistema se podían utilizar lenguajes de programación concurrente como el Pascal

Concurrente, Modula, Ada, etc. BACI, sin embargo, está específicamente diseñado para el desarrollo de este tipo de prácticas, por lo que el alumno se distrae menos del objetivo del entendimiento de la programación concurrente, con detalles particulares del sistema operativo.

Otro modo de plantear la aplicación de la programación concurrente es mediante la utilización de los mecanismos IPC que ofrece el Sistema Operativo UNIX, utilizados en la resolución de un problema que conlleve la necesidad de varios procesos que cooperen entre sí y se sincronicen. Las Palmas de Gran Canaria es un ejemplo de ello.

4. Windows: ¿un advenedizo indeseable?

Otra consideración que nos parece interesante resaltar es la ausencia casi total del sistema operativo Windows en las prácticas de Sistemas Operativos.

Esta ausencia es justificable en algunos tipos de prácticas:

- En las prácticas a nivel de usuario del sistema suele elegirse Unix porque, normalmente, el manejo de los sistemas Windows es muy intuitivo, es conocido por los alumnos y ya ha sido introducido en alguna otra asignatura de primer curso al usarse ese entorno en sus prácticas.
- En las prácticas a nivel de diseñador del sistema, cuando se trata de modificar algún sistema existente, es indispensable utilizar un sistema del que se disponga el código fuente, cosa que, obviamente, no ocurre en Windows.

En el resto de prácticas que se han enumerado en el apartado 3, no hay ninguna razón inherente a la práctica que justifique la utilización de un sistema u otro.

Normalmente suele justificarse la elección de Unix (o Linux) porque es mucho mejor conocido, tanto internamente como a nivel de programador o usuario que el Windows. Además, existe una cierta tendencia en muchos informáticos a preferir entornos más “áridos” para marcar diferencias con el resto de usuarios (“es que Windows lo maneja cualquiera”). Si a esto unimos la difusión de Linux y su carácter gratuito, y el “romanticismo” que conlleva el enfrentamiento al gigante Microsoft, hace que muchos docentes prefieran

desechar este sistema para las prácticas de Sistemas Operativos.

Sin embargo, es un hecho evidente que el sistema más implantado, y con mucha diferencia, es alguno de los sistemas Windows, con lo que la mayoría de los alumnos deberán enfrentarse a este sistema en su vida laboral. Parece, por tanto, prudente incidir en el conocimiento de este sistema operativo, tanto en el apartado de teoría como en el de prácticas, sin que esto suponga abandonar por completo el uso de Unix.

Esta utilización de Windows tiene mucho mayor sentido si queremos dar a nuestra asignatura un carácter finalista, incluyendo esta materia que podrá ser utilizada por los alumnos en su posterior vida laboral.

En contra de esta inclusión puede justificarse que tal vez es excesivo utilizar dos conjuntos de llamadas al sistema (Unix y Windows), con lo cual no se logrará profundizar en el conocimiento de ninguno de los dos. Puede ser cierto, pero consideramos que lo más difícil y que más tiempo puede llevar a la hora de realizar programas que utilicen una API determinada es empezar a utilizarla. Una vez conocidos los aspectos básicos de funcionamiento, los conceptos que utiliza y los tipos de datos implicados, el conocer más o menos funciones concretas es cuestión de tiempo, no de dificultad.

Por lo tanto, lo que proponemos en nuestras prácticas es sentar las bases para la utilización de las llamadas al sistema mediante la implementación de un programa del sistema típico, donde se utilizan las llamadas más importantes. Una vez hecho esto, el alumno, que por interés o necesidad quiera profundizar más en uno de los sistemas que utilizamos lo tendrá mucho más fácil.

Además, hay que considerar el hecho que la mayor parte del tiempo de realización de esta práctica se lo va a llevar la lógica del programa, no los problemas que puedan surgir con las llamadas al sistema. Al ser el problema a resolver el mismo en Unix y en Windows se ahorra mucho tiempo que se dedica a estudiar en detalle las correspondientes APIs de los sistemas.

5. Nuestra propuesta de prácticas

A la hora de diseñar nuestro programa de prácticas hemos tenido en cuenta todas las reflexiones anteriormente señaladas. En este punto

describiremos tanto los objetivos que pretendemos lograr como el esquema de prácticas que hemos planteado.

5.1. Objetivos

Las prácticas de laboratorio, persiguen una doble finalidad: por un lado deben ser un complemento de los aspectos teóricos estudiados en las clases de teoría; por otro lado, deben permitir fijar y comprender mejor los sistemas operativos en los tres niveles propuestos como objetivo: usuario, programador y diseñador del sistema.

Por lo tanto, el alumno trabajará y adquirirá destreza y conocimientos acerca de los sistemas operativos en tres niveles diferentes a través de esta asignatura:

1. Nivel de usuario, mediante el cuál el alumno reconoce, utiliza y diferencia las herramientas de alto nivel que acompañan al sistema operativo
2. Nivel de programador de aplicaciones que proporciona al alumno una visión directa y global de las funciones que el Sistema Operativo ofrece a través de su interfaz de llamadas al sistema
3. Nivel de diseñador, a través del cuál el alumno descubre, desarrolla y relaciona las funciones y mecanismos que proporciona el sistema operativo

Como objetivos específicos podemos señalar:

- Conocer los servicios que ofrecen los sistemas operativos a los distintos tipos de usuarios: usuario de aplicaciones, usuario programador y administrador del sistema.
- Adquirir destreza en su uso y percibir su utilidad en diferentes campos.
- Ser capaces de implementar algunos de los subsistemas que constituyen un sistema operativo.

La ponderación de los objetivos que cubren cada práctica en relación al total de objetivos prácticos que hemos establecido es la siguiente:

- Conocimientos a nivel de usuario 15%
- Conocimientos a nivel de programación del sistema 35%
- Conocimientos a nivel de diseño interno del sistema 50%

Esta ponderación marca el tiempo empleado en cada nivel y también la evaluación de la práctica.

De los tipos de prácticas que hemos comentado en el punto 3 no hemos incluido nada relativo al tema de programación concurrente, dado que consideramos que este tema no debe aparecer en la asignatura de sistemas operativos [30].

5.2. Contenido de las prácticas

A partir del análisis anterior de los tipos de prácticas y teniendo en cuenta las circunstancias propias que rodean esta asignatura, así como los objetivos generales y específicos de la asignatura, se ha elaborado un plan mixto de prácticas con el siguiente contenido:

- En primer lugar se realiza un módulo en el que el alumno se introduzca en el *manejo del Sistema Operativo UNIX*, nuevo para él, por lo que necesita desenvolverse al nivel de usuario. No se considera necesarios incluir el estudio de Windows en este módulo dado que ya ha sido estudiado como base para las prácticas en las asignaturas de primer curso.
- Una vez que el alumno adquiere cierta soltura con la máquina, la segunda práctica introduce al alumno en los aspectos de uso del sistema operativo como programador de aplicaciones. El concepto de "*llamadas al sistema*", estudiado en la teoría se pone en práctica con el uso de llamadas para la creación de procesos, comunicación, etc, para lo cual es necesario comprender perfectamente el modo en que actúa internamente tanto *el Sistema Operativo UNIX* (en su versión libre de Linux), como el *Sistema Operativo Windows XP* (o algún otro basado en el núcleo de NT).
- Finalmente se introduce un módulo en el que se aplicarán los conceptos vistos en las clases teóricas, relativos al diseño de las funciones internas del sistema operativo. En este módulo *serán simulados los bloques básicos del Sistema Operativo: procesos, memoria, ficheros y E/S*. El objetivo es utilizar las principales estructuras de datos propias del sistema operativo e implementar su manejo, siempre de un modo simulado, eliminando la complejidad de un diseño sobre una plataforma real.

Teniendo en cuenta todo lo expuesto, se proponen 5 prácticas, donde la primera corresponde al

primer módulo, la 2 y la 3 al segundo, mientras que la 4 y la 5 se incluyen en el tercero:

1. Introducción, a nivel del usuario de órdenes, al S.O. Unix
2. Uso de la Interfaz de Programas de Aplicación del estándar POSIX.
3. Uso de la Interfaz de Programas de Aplicación de Win32.
4. Simulación de Gestión de Procesos y Memoria.
5. Simulación de la Gestión de ficheros y entrada/salida a disco.

El entorno de desarrollo a utilizar en las prácticas 4 y 5 puede ser, indiferentemente, sistemas Unix o sistemas Windows. En nuestro caso, además, se realizará utilizando el lenguaje Java, dado que es el que mejor conocen los alumnos. Las prácticas 2 y 3 se realizarán en lenguaje C.

Este programa de prácticas se ha establecido teniendo en cuenta que la carga de prácticas de la asignatura es de 3 créditos, lo que se traduce en una hora de prácticas semanales durante todo el curso. La práctica 1 dura 4 semanas, las práctica 2 y 3 se prolongarán durante 5 semanas, mientras que las dos restantes tienen prevista una duración de 8 semanas cada una de ellas.

6. La opinión del alumno

Considerando que es el primer año en el que se ha aplicado el esquema de prácticas anteriormente expuesto, los resultados obtenidos son necesariamente limitados. A lo largo del curso se ha mantenido un diálogo constante con los alumnos para ir recabando sus impresiones en relación a éste y a otros aspectos de la asignatura. En cualquier caso son opiniones subjetivas, tanto las expresadas por los alumnos como la impresión que de ellas obtienen los profesores, pero de estas conversaciones podemos destacar las siguientes impresiones:

- Lo más complicado para el alumno ha sido el adaptarse al entorno de trabajo en Unix, dado que para la mayoría es la primera vez que lo utilizan.
- No se han encontrado problemas especiales en la utilización de dos lenguajes de programación. Java lo dominan los alumnos perfectamente puesto que es primer lenguaje que aprenden en la carrera, mientras que C lo aprenden en otra asignatura de segundo con anterioridad a necesitarlo en la asignatura de Sistemas Operativos, y este lenguajes es

considerablemente más simple que Java. Se ha destacado, además, cómo han sido muy útiles las prácticas de programación en C en nuestra asignatura para reforzar el conocimiento general de este lenguaje.

- La inclusión de programación con llamadas al sistema de Windows ha sido valorada muy positivamente, dado que éste es el sistema más utilizado en la actualidad en general y por los alumnos en particular.
- El que la práctica 3 sea, de hecho, una migración de la práctica 2 a un sistema distinto también ha sido bien visto por los alumnos, por dos razones distintas. En primer lugar, facilita mucho la realización de la práctica 3, al poder reutilizar casi todo el código de la práctica 2. En segundo lugar, algunos alumnos han resaltado el hecho de que el realizar migraciones entre sistemas distintos son habituales en la vida profesional del informático, y este es una buena práctica de ello.
- La realización de las prácticas de simulación de partes del sistema operativo les ha resultado muy útil para resolver dudas y profundizar en el conocimiento de lo explicado en teoría.

Como punto negativo expresado por el alumno es el excesivo número de prácticas que tienen que realizar durante el curso, dado que prácticamente todas las asignaturas tienen prácticas de laboratorio. Este aspecto negativo no es provocado por la asignatura en sí (no se considera excesivo el trabajo solicitado), sino por la gran carga de prácticas que tiene el plan de estudios en general.

7. Conclusiones

Haciendo una revisión exhaustiva de los programas de las asignaturas relacionadas con Sistemas Operativos nos hemos encontrado con que hay una gran variedad de objetivos y contenidos.

Siendo todos ellos aceptables desde algún punto de vista, consideramos que tratados aisladamente no completan todos los aspectos que deberían ser tratados en la asignatura.

Otra carencia importante que se ha detectado es la casi completa ausencia del sistema operativo Windows, en alguna de sus versiones, de los programas de prácticas. Creemos que este sistema

tiene la suficiente importancia hoy en día para ser incluido en las prácticas de Sistemas Operativos.

Teniendo en cuenta todo lo anterior, se ha diseñado un programa de prácticas que pretende por un lado desarrollar las habilidades del alumno como usuario de órdenes y de aplicaciones, tanto de sistemas tipo Unix como sistemas Windows, mientras que por otro lado quiere profundizar en los conocimientos necesarios para el diseñador de sistemas operativos.

Este programa ha sido puesto en práctica durante este curso, coincidiendo con el cambio de Plan de Estudios, y se han obtenido dos conclusiones:

- Los alumnos adquieren soltura en el manejo de los sistemas Windows y Unix, tanto a nivel de usuario de órdenes como de programador.
- La simulación de partes del sistema operativo ayuda considerablemente a su comprensión por parte del alumno.

Por último, hay que constatar la agradable sorpresa que ha supuesto para los profesores la calidad de muchas de las prácticas presentadas, sobre todo en las prácticas 4 y 5. El hecho de haber cursado la asignatura “Comunicación Persona-Máquina” en primer curso ha hecho que muchas de las prácticas presenten un rico y cuidado interfaz gráfico de usuario que, unido a su buen funcionamiento interno, hacen de estas prácticas muy buenas herramientas de apoyo a la docencia de partes determinadas del programa de teoría.

Referencias

- [1] *Asignatura de Diseño de Sistemas Operativos de la EPSI de Gijón, Universidad de Oviedo*, <http://www.epsig.uniovi.es/asignaturas/dso/index.html>
- [2] *Asignatura de Sistemas Operativos en la Universidad de Duke*, <http://www.cs.duke.edu/courses/spring03/cps210/>
- [3] *Asignatura de Sistemas Operativos en la Universidad de Granada*, <http://lsi.ugr.es/~jagomez/sisopi.html#Temario>
- [4] *Asignatura de Sistemas Operativos en la EUIT en Informática de Gijón, Universidad de Oviedo*, <http://www.inforg.uniovi.es/so/>
- [5] *Asignatura de Sistemas Operativos en la Universidad Carlos III de Madrid*, <http://arcos.inf.uc3m.es/~ssoo/descripcion.html>
- [6] *Asignatura de Sistemas Operativos en la Universidad Brown*, <http://www.cs.brown.edu/courses/cs167/>
- [7] *Asignatura de Sistemas Operativos en la Universidad de Chigaco*, <http://people.cs.uchicago.edu/~odonnell/OData/Courses/CS230/contents.html>
- [8] *Asignatura de Sistemas Operativos en la Universidad de Harvard*, <http://das-www.harvard.edu/cs/academics/courses/cs161/cs161.html>
- [9] *Asignatura de Sistemas Operativos en la Universidad de Oregón*, <http://www.cs.uoregon.edu/classes/04W/cis415/cis415/cis415.html>
- [10] *Asignatura de Sistemas Operativos en la Universidad de Pennsylvania State*, <http://www.cse.psu.edu/~anand/411/411.html>
- [11] *Asignatura de Sistemas Operativos en la Universidad de Sevilla*, http://www.lsi.us.es/docencia/asignaturas/so_inti.html#tem
- [12] *Asignatura de Sistemas Operativos en la Universidad de Valladolid*, <http://www.infor.uva.es/~isaac/SO/so.html>
- [13] *Asignatura de Sistemas Operativos en la Universidad de Vrije*, <http://www.cs.vu.nl/~steen/bs/>
- [14] *Asignatura de Sistemas Operativos en la Universidad de Wisconsin*, <http://www.cs.wisc.edu/~cs537-1/>
- [15] *Asignatura de Sistemas Operativos en la Universidad Politécnica de Cataluña*, <http://studies.ac.upc.es/FIB/SO/>
- [16] *Asignatura de Sistemas Operativos en la Universidad Politécnica de Madrid*, <http://www.if.upm.es>
- [17] Bynum B. y Camp, T., *An Introduction to BACI*, http://www.mines.edu/fs_home/tcamp/baci/
- [18] Carretero Pérez, J. y otros. *Prácticas de Sistemas Operativos. De la base al diseño Una visión aplicada*. McGraw Hill, 2002
- [19] Carretero Pérez, J. y otros. *Sistemas Operativos. Una visión aplicada*. McGraw Hill, 2001
- [20] Christopher, W., S. Procter, T. Anderson *The Nachos Instructional Operating System*.

- Proceedings of the 1993 Winter USENIX Conference. 1993
- [21] Comer, Douglas Operating System Design, the XINU approach Prentice Hall. 1984
- [22] Consejo de Universidades. *Directrices generales propias de los planes de estudios conducentes a la obtención del título oficial de Ingeniero en Informática*. Boletín Oficial del Estado, número 278, pag. 34401 a 34403
- [23] Consejo de Universidades. *Directrices generales propias de los planes de estudios conducentes a la obtención del título oficial de Ingeniero Técnico en Informática de Gestión*. Boletín Oficial del Estado, número 278, pag. 34403 y 34404
- [24] Consejo de Universidades. *Directrices generales propias de los planes de estudios conducentes a la obtención del título oficial de Ingeniero Técnico en Informática de Sistemas*. Boletín Oficial del Estado, número 278, pag. 34404 y 34405
- [25] Deitel, H. M. *Sistemas Operativos. Segunda Edición*. Addison-Wesley Iberoamericana, 1993
- [26] Dickinson, J. *Operating systems projects built on a simple hardware simulator*, Proceedings of the 31 SIGCSE technical symposium on Computer science education, ACM Press, 2000
- [27] Linux Home Page <http://www.linux.org/>
- [28] Minix <http://www.cs.vu.nl/~ast/minix.html>
- [29] Nachos Home Page
<http://http.cs.berkeley.edu/~tea/nachos/>
- [30] Riesco Albizu, M. y Díaz Fondón, M.A., *La Programación Concurrente y el Interbloqueo en la Asignatura de Sistemas Operativos*, Actas de las IX Jornadas de Enseñanza Universitaria de la Informática, Ed. Thomsom, 2003.
- [31] Sinlverschatz, A. y Galván, P.B. *Sistemas Operativos*. Quinta Edición. Addison Wesley Longman, 1999.
- [32] Stallings, W. *Sistemas Operativos*. Cuarta Edición. Prentice Hall, 2001.
- [33] Switzer, R. *Operating Systems: A Practical Approach*, Prentice-Hall, 1993
- [34] Tanenbaum, A. y Woodhull A. *Operating Systems Design and Implementation*, 2ª edición. Prentice-Hall, 1997
- [35] The Joint Task Force on Computing Curricula, *Computing Curricula 2001* IEEE Computer Society, Association for Computing Machinery, 2000
- [36] UNESCO/IFIP, *A Modular Curriculum in Computer Science*. UNESCO, 1994.