26-27

MICROGRADO EN DISEÑO Y PROGRAMACIÓN DE APLICACIONES DISTRIBUIDAS
OPTATIVAS CURSO

# GUÍA DE ESTUDIO PÚBLICA

# PROGRAMACIÓN ORIENTADA A OBJETOS

**CÓDIGO 71901072**

UNED

26-27

**PROGRAMACIÓN ORIENTADA A OBJETOS**

**CÓDIGO 71901072**

# ÍNDICE

PRESENTACIÓN Y CONTEXTUALIZACIÓN
REQUISITOS Y/O RECOMENDACIONES PARA CURSAR LA ASIGNATURA
EQUIPO DOCENTE
HORARIO DE ATENCIÓN AL ESTUDIANTE
TUTORIZACIÓN EN CENTROS ASOCIADOS
COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE
RESULTADOS DE APRENDIZAJE
CONTENIDOS
METODOLOGÍA
SISTEMA DE EVALUACIÓN
BIBLIOGRAFÍA BÁSICA
BIBLIOGRAFÍA COMPLEMENTARIA
RECURSOS DE APOYO Y WEBGRAFÍA
IGUALDAD DE GÉNERO

| | |
|---|---|
| NOMBRE DE LA ASIGNATURA | PROGRAMACIÓN ORIENTADA A OBJETOS |
| CÓDIGO | 71901072 |
| CURSO ACADÉMICO | 2026/2027 |
| DEPARTAMENTO | LENGUAJES Y SISTEMAS INFORMÁTICOS |
| TÍTULO EN QUE SE IMPARTE | GRADO EN INGENIERÍA INFORMÁTICA |
| CURSO - PERIODO - TIPO | - PRIMER<br>- SEMESTRE 2<br>- FORMACIÓN BÁSICA |
| TÍTULO EN QUE SE IMPARTE | MICROGRADO EN PROGRAMACIÓN AVANZADA DE ALGORITMOS |
| PERIODO - TIPO | - SEMESTRE 2<br>- OBLIGATORIAS |
| TÍTULO EN QUE SE IMPARTE | MICROGRADO EN DISEÑO Y PROGRAMACIÓN DE APLICACIONES DISTRIBUIDAS |
| PERIODO - TIPO | - SEMESTRE 2<br>- OBLIGATORIAS |
| TÍTULO EN QUE SE IMPARTE | GRADO EN INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN |
| CURSO - PERIODO - TIPO | - PRIMER<br>- SEMESTRE 2<br>- FORMACIÓN BÁSICA |
| TÍTULO EN QUE SE IMPARTE | MÁSTER UNIVERSITARIO EN INGENIERÍA DE LAS TECNOLOGÍAS EDUCATIVAS (COMPLEMENTO) |
| Nº ETCS | 6 |
| HORAS | 150.0 |
| IDIOMAS EN QUE SE IMPARTE | CASTELLANO |

## PRESENTACIÓN Y CONTEXTUALIZACIÓN

El objetivo de esta guía es orientar al alumno en el estudio de la asignatura. Se recomienda la lectura completa de la guía a comienzo del cuatrimestre para tener una idea completa de la temática de la asignatura y el calendario de prácticas, de forma que el alumno pueda planificar su trabajo para cumplir con las fechas de entrega.

La asignatura de Programación Orientada a Objetos (POO en adelante) se imparte en el segundo cuatrimestre del primer curso, consta de 6 créditos, con carácter de formación básica, para la titulación de grado en Informática. Y en el grado de Tecnologías de la Información, la asignatura también se imparte en el segundo cuatrimestre del primer curso, consta de 6 créditos y tiene carácter de formación básica.

La asignatura Programación Orientada a Objetos se engloba dentro de la materia de Programación.

Esta asignatura es una introducción a la POO y tiene como objetivo principal que el alumno adquiera los conocimientos y habilidades necesarios para poder desarrollar programas básicos utilizando la metodología de la orientación a objetos. Dado el hecho de que es una asignatura de primer curso, en lugar de emplear un entorno estándar de desarrollo de software como (por ejemplo) Eclipse, se usa BlueJ debido a su naturaleza didáctica.

Se trata de una continuación de la asignatura Fundamentos de Programación que se centra en los tipos abstractos de datos. En Fundamentos los alumnos ven un lenguaje híbrido C±, que mezcla partes de los lenguajes de programación C y C++, pero (en términos de la abstracción de datos) que no va más allá que la utilización de la abstracción STRUCT. Así que esta asignatura, que presenta mecanismos de la POO como la herencia y el polimorfismo, representa una extensión natural de aquélla. El hecho de haber cursado Fundamentos ayudará a los estudiantes a entender muchos conceptos de la POO.
Tanto el lenguaje de programación tratado en esta asignatura, Java, como el paradigma de programación orientada a objetos, son de gran utilidad para los alumnos en el mundo profesional de la informática actual y son muy demandados.

## REQUISITOS Y/O RECOMENDACIONES PARA CURSAR LA ASIGNATURA

Al tratarse de una asignatura de carácter introductorio, no se requiere ningún requisito previo más allá de los conocimientos que un alumno debe tener en este nivel de enseñanza y del manejo de un ordenador personal a nivel de un usuario doméstico.

## EQUIPO DOCENTE


| | |
|---|---|
| Nombre y Apellidos | TIMOTHY MARTIN READ (Coordinador/a de asignatura) |
| Correo Electrónico | tread@lsi.uned.es |
| Teléfono | 91398-8261 |
| Facultad | ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA |
| Departamento | LENGUAJES Y SISTEMAS INFORMÁTICOS |

| | |
|---|---|
| Nombre y Apellidos | JUAN MARTINEZ ROMO |
| Correo Electrónico | juaner@lsi.uned.es |
| Teléfono | 91398-9378 |
| Facultad | ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA |
| Departamento | LENGUAJES Y SISTEMAS INFORMÁTICOS |

| | |
|---|---|
| Nombre y Apellidos | JOSE LUIS DELGADO LEAL |
| Correo Electrónico | jdelgado@lsi.uned.es |
| Teléfono | 91398-8736 |
| Facultad | ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA |
| Departamento | LENGUAJES Y SISTEMAS INFORMÁTICOS |

| | |
|---|---|
| Nombre y Apellidos | ANDRES DUQUE FERNANDEZ |
| Correo Electrónico | aduque@lsi.uned.es |
| Teléfono | 91398-6535 |
| Facultad | ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA |
| Departamento | LENGUAJES Y SISTEMAS INFORMÁTICOS |

| | |
|---|---|
| Nombre y Apellidos | MIGUEL ANGEL RODRIGUEZ GARCIA |
| Correo Electrónico | miguelangel.rodriguez@lsi.uned.es |
| Teléfono | 91398-9669 |
| Facultad | ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA |
| Departamento | LENGUAJES Y SISTEMAS INFORMÁTICOS |

## HORARIO DE ATENCIÓN AL ESTUDIANTE

Datos de contacto y horario de asistencia al estudiante:

•Profesor Timothy Read

Miércoles de 09:30 a 13:30 horas.

Despacho 2.01 de la ETSI Informática tfno: 91 398 8260

•Profesor José Luis Delgado Leal:

Jueves, de 15:00 horas a 19:00 horas.

Despacho 2.13 de la ETSI Informática tfno: 91 398 87 36

•Profesor Juan Martínez Romo:

Martes y jueves de 11:00 a 13:00 horas.

Despacho 2.03 de la ETSI Informática tfno: 91 398 93 78

•Profesor Andrés Duque Fernández:

Jueves de 11:00 a 13:00 horas y de 15:00 a 17:00 horas.

Despacho 2.06 de la ETSI Informática tfno: 91 398 6535

Dirección postal:

Equipo docente de la asignatura Programación Orientada a Objetos

Departamento de Lenguajes y Sistemas Informáticos

ETSI Informática

C./ de Juan del Rosal, 14

28040 Madrid

## TUTORIZACIÓN EN CENTROS ASOCIADOS

En el enlace que aparece a continuación se muestran los centros asociados y extensiones en las que se imparten tutorías de la asignatura. Estas pueden ser:

•**Tutorías de centro o presenciales**: se puede asistir físicamente en un aula o despacho del centro asociado.

•**Tutorías campus/intercampus**: se puede acceder vía internet.

Consultar horarios de tutorización de la asignatura 71901072

## COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE

Con el estudio de la asignatura el alumno debe adquirir una visión general de la POO. La POO es una evolución de la programación procedural basada en funciones. La POO nos permite agrupar secciones de código con funcionalidades comunes. Una de las principales desventajas de la programación procedural basada en funciones es su construcción, cuando una aplicación bajo este tipo de programación crece, la modificación del código se hace muy trabajosa y difícil debido a que el cambio de una sola línea en una función, puede acarrear la modificación de muchas otras líneas de código pertenecientes a otras funciones que estén relacionadas.
Con la programación orientada a objetos se pretende agrupar el código encapsulándolo y haciéndolo independiente, de manera que una modificación debida al crecimiento de la aplicación solo afecte a unas pocas líneas. La organización de una aplicación en POO se realiza mediante estructuras de código, también llamados objetos. Estos objetos contienen una serie de procedimientos e información destinados a resolver un grupo de tareas con un denominador común. Un procedimiento que este situado en un objeto no podrá ser usado por otro procedimiento perteneciente a otro objeto, si no es bajo una serie de reglas. Los datos que mantenga el objeto, permanecerán aislados del exterior y sólo se podrá acceder a ellos siguiendo ciertas normas. El objetivo de POO es catalogar y diferenciar el código, en base a estructuras jerárquicas dependientes, al estilo de un árbol genealógico.
Las competencias adquiridas en la asignatura son:

(G.2) Competencias cognitivas superiores: selección y manejo adecuado de conocimientos, recursos y estrategias cognitivas de nivel superior apropiados para el afrontamiento y resolución de diversos tipos de tareas/problemas con distinto nivel de complejidad y novedad: Análisis y Síntesis. Aplicación de los conocimientos a la práctica Resolución de problemas en entornos nuevos o poco conocidos. Pensamiento creativo. Razonamiento crítico. Toma de decisiones.

(G.4) Competencias de expresión y comunicación (a través de distintos medios y con distinto tipo de interlocutores): Comunicación y expresión escrita. Comunicación y expresión oral. Comunicación y expresión en otras lenguas (con especial énfasis en el inglés). Comunicación y expresión matemática, científica y tecnológica (cuando sea requerido y estableciendo los niveles oportunos).

(FB.5) Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, así como de los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

(BC.1) Capacidad para diseñar, desarrollar, seleccionar y evaluar, aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a los principios éticos y a la legislación y normativa vigente.

(BTEisw.1) Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.

## RESULTADOS DE APRENDIZAJE

Con el estudio de la asignatura el alumno debe adquirir una visión general de la POO. La POO es una evolución de la programación procedural basada en funciones. La POO nos permite agrupar secciones de código con funcionalidades comunes. Una de las principales desventajas de la programación procedural basada en funciones es su construcción, cuando una aplicación bajo este tipo de programación crece, la modificación del código se hace muy trabajosa y difícil debido a que el cambio de una sola línea en una función, puede acarrear la modificación de muchas otras líneas de código pertenecientes a otras funciones que estén relacionadas.
Con la programación orientada a objetos se pretende agrupar el código encapsulándolo y haciéndolo independiente, de manera que una modificación debida al crecimiento de la aplicación solo afecte a unas pocas líneas. La organización de una aplicación en POO se realiza mediante estructuras de código, también llamados objetos. Estos objetos contienen una serie de procedimientos e información destinados a resolver un grupo de tareas con un denominador común. Un procedimiento que este situado en un objeto no podrá ser usado por otro procedimiento perteneciente a otro objeto, si no es bajo una serie de reglas. Los datos que mantenga el objeto, permanecerán aislados del exterior y sólo se podrá acceder a ellos siguiendo ciertas normas. El objetivo de POO es catalogar y diferenciar el código, en base a estructuras jerárquicas dependientes, al estilo de un árbol genealógico.
Los resultados de aprendizaje para las competencias adquiridas en la asignatura son:

R4. Realizar pruebas de validación y depuración de un programa dado (competencias: G.2, FB.5, BC.1).
R5. Identificar clases de objetos con los datos de un problema (competencias: G.2, G.4, BC.1, FB.5, BTEisw.4)
R6. Crear clases y objetos y manipularlos (competencias: G.2, FB.5, BC.1)
R7. Identificar los métodos que caracterizan a una clase de objetos (competencias: G.2, FB.5, BC.1)
R8. Utilizar el mecanismo de la herencia para crear clases (competencias: G.2, FB.5, BC.1)
R9. Abordar metodológicamente proyectos de programación sencillos (competencias: G.2, FB.5, BC.1)
R10. Dominar un lenguaje de programación acorde al paradigma del contenido de la materia (competencias: G.2, FB.5, BC.1)

## CONTENIDOS

Parte I. Fundamentos de programación orientada a objetos

Tema 1. Objetos y clases
Tema 2. Definiciones de clases
Tema 3. Interacción de objetos
Tema 4. Agrupación de objetos
Tema 6. Comportamientos más sofisticados
Tema 7. Colecciones de tamaño fijo: matrices

Parte 2. Estructuras de las aplicaciones

Tema 8. Diseño de clases
Tema 9. Objetos con un buen comportamiento
Tema 10. Mejora de la estructura mediante la herencia
Tema 11. Más sobre la herencia
Tema 12. Técnicas de abstracción adicionales
Tema 14. Tratamiento de errores

## METODOLOGÍA

La enseñanza a distancia posee unas características que la diferencian de la presencial. Sin embargo, esto no impide que el alumno pueda disponer de la ayuda y los recursos necesarios para cursar las asignaturas en las que se matricule.
Los mecanismos que tiene el alumno para facilitar el aprendizaje requerido en la asignatura son los siguientes:

•Tutorías presenciales o virtuales en el centro asociado correspondiente. Se puede consultar los horarios de tutorización de la asignatura en el siguiente enlace:
https://akademosweb.uned.es/Default.aspx?id_servicio=19&modo_23=1&asigna=71901072
Además de tutorizar los contenidos teóricos de la asignatura, los tutores serán los encargados de desarrollar las sesiones presenciales de seguimiento y control de las prácticas. Para la realización de las prácticas el alumno debe ponerse en contacto con el tutor correspondiente, para conocer cuanto antes el horario y las sesiones previstas.

•Entorno Virtual. A través de la plataforma educativa, el equipo docente de la asignatura proporcionará a los alumnos una variedad de materiales de apoyo para el estudio, así como el enunciado de la práctica obligatoria. También se ofrecen foros donde los alumnos pueden plantear sus preguntas y recibir respuestas de los tutores o del equipo docente. El entorno virtual es el soporte fundamental de la asignatura y la principal herramienta de comunicación entre el equipo docente, los tutores y los alumnos, así como entre los propios

alumnos.

La modalidad y tipo de actividades que se contemplan incluye: trabajo con contenidos teórico-prácticos utilizando la bibliografía y el material complementario. Trabajo autónomo con las actividades de ejercicios y pruebas de autoevaluación disponibles, y realización de una práctica bajo la supervisión del tutor, con las herramientas y directrices preparadas por el equipo docente. La interacción con el equipo docente y los tutores se describe más adelante.

Para solicitar plaza/turno de prácticas de laboratorio/experimentales, el estudiante tendrá que acceder a la aplicación de prácticas desde su escritorio. En estas imágenes puede ver desde dónde se puede realizar el acceso a dicha aplicación:
https://descargas.uned.es/publico/pdf/guias/ACCESO_PRACTICAS_GRADOS_2017.pdf
Si al acceder a ella no encuentra ninguna oferta, deberá ponerse en contacto con el centro asociado donde está matriculado.

Finalmente, cabe destacar que el libro base de la asignatura tiene una estructura muy didáctica, que se ha desarrollado a lo largo de muchas ediciones, que puede facilitar mucho el estudio del alumno. También, hay más recursos disponibles al alumno en el curso virtual de la asignatura.

## SISTEMA DE EVALUACIÓN

**TIPO DE PRUEBA PRESENCIAL**

| | |
|---|---|
| Tipo de examen | Examen mixto |
| Preguntas test | 16 |
| Preguntas desarrollo | 4 |
| Duración del examen | 120 (minutos) |

Material permitido en el examen

No está permitido el uso de material adicional.

Criterios de evaluación

La evaluación de la asignatura se basa en el desarrollo de una práctica obligatoria y un examen final. La nota final de la asignatura se obtendrá mediante la influencia de tres partes:

**Práctica Obligatoria**:

Los alumnos que hayan aprobado la práctica en el curso anterior, no tienen que realizar la práctica de este año, ya que se guarda la nota del año pasado.

No obstante, las preguntas en la parte de desarrollo del examen irán sobre la práctica de este año así que los alumnos deberían estudiar el enunciado de la práctica de este curso, independiente de si van a realizar la práctica o no.

Los alumnos que no puedan entregar la práctica para examinarse en junio pueden entregarla más adelante para examinarse en septiembre. Si el tutor en su centro asociado acepta corregir prácticas para septiembre y propone una segunda fecha, los alumnos del centro asociado correspondiente tienen que entregar la práctica al tutor en la fecha que este les marque.

**Los alumnos que no disponen de un tutor en su centro asociado que pueda corregir prácticas para septiembre deben mandar sus prácticas por correo electrónico a otro tutor cuyo nombre pondrá el equipo docente en el foro más adelante. Para poder pedir la corrección de la práctica ese tutor es imprescindible la asistencia a la sesión de control (confirmada por el tutor de su centro asociado).**

**Parte Teórica del Examen Final**: La parte teórica consta de 16 preguntas tipo test (14 oficiales + 2 de reserva que hay que contestar). El contenido de las preguntas versa sobre la totalidad del temario de la asignatura y es requisito necesario su superación. Para aprobarlo será necesario obtener en esta parte al menos una calificación de 1,25 puntos sobre la nota máxima del mismo, que es de 2,5 puntos (lo que equivale a un 50% del total o 5 puntos sobre 10 puntos si consideramos la nota ponderada a una escala entre 0 y 10). Si no se aprueba la parte teórica (test), la parte práctica del examen final no será corregida.

**Aunque en el ánimo del Equipo Docente está en la medida de lo posible evitar posibles errores de diversa índole en la generación de las preguntas del test (llevando un riguroso control de la preparación de las diferentes cuestiones con el fin de evitar erratas en las mismas), a veces se producen estos errores. Para minimizar los efectos de dichos errores, se ha establecido que la lógica de proceder será la siguiente:**

El examen constará de un total de 16 preguntas de tipo test.

De estas 16 preguntas, 14 preguntas (que aparecerán indicadas en el examen) serán las que sirvan como base para obtener la calificación final del test, quedando las otras dos preguntas como preguntas de reserva.

En caso de producirse un error objetivo en la formulación de la pregunta y/o en las posibles respuestas alternativas de la pregunta, automáticamente se procederá a anular dicha pregunta, y en su lugar se tomará en consideración la primera de las preguntas de reserva, para que de este modo el examen siga estando conformado por un total de 14 preguntas.

Si, desafortunadamente, volviese a ocurrir esta circunstancia una segunda vez, se repetiría el proceso de anulación de la pregunta erro¿nea, y se tomaría en ese caso la segunda pregunta de reserva establecida a tal fin.
A partir de este punto, si excepcionalmente ocurriese este hecho una tercera vez (o sucesivas), la preguntas erróneas se considerarían como correctas (exclusivamente esa tercera pregunta o posteriores) para todos los alumnos que hayan realizado el examen.

**Parte Práctica del Examen**: La temática de la parte práctica (desarrollo) estará intrínsecamente relacionada con la práctica obligatoria realizada por el alumno a lo largo del curso. La nota máxima es de 6,5 puntos.

| | |
|---|---|
| % del examen sobre la nota final | 90 |
| Nota del examen para aprobar sin PEC | |
| Nota máxima que aporta el examen a la calificación final sin PEC | |
| Nota mínima en el examen para sumar la PEC | |
| Comentarios y observaciones | |

**PRUEBAS DE EVALUACIÓN CONTINUA (PEC)**

| | |
|---|---|
| ¿Hay PEC? | No |
| Descripción | |
| Criterios de evaluación | |
| Ponderación de la PEC en la nota final | |
| Fecha aproximada de entrega | |
| Comentarios y observaciones | |

**OTRAS ACTIVIDADES EVALUABLES**

| | |
|---|---|
| ¿Hay otra/s actividad/es evaluable/s? | Si |

Descripción

**Práctica Obligatoria**

**Los objetivos que se plantean en la realización de esta práctica son los siguientes:**

Familiarización con la Programación Orientada a Objetos (POO): definición de clases e instancias, uso de la herencia, definición/uso de métodos estáticos y abstractos.

Realización del diseño orientado a objetos de un problema.

Implementacio¿n de un programa sencillo donde se manejen conceptos relacionados con POO.

**La práctica se va a implementar en Java 2 Estándar Edition (J2SE). El compilador de Java que se usará será BlueJ, tal y como se define en el programa de la asignatura.**

Criterios de evaluación

La realización de la práctica es obligatoria. Sólo se evaluará el examen si la práctica ha sido previamente aprobada.
Aunque si bien el desarrollo de aplicaciones Orientadas a Objetos usando el lenguaje de programación Java no requiere el uso concreto de ningún entorno de desarrollo, está práctica ha de desarrollarse íntegramente empleando el entorno de desarrollo BlueJ, que es el que se muestra en el libro de texto básico de la asignatura.
La práctica es un trabajo individual. Las prácticas cuyo código coincida total o parcialmente con el de otro alumno serán motivo de suspenso para todos los implicados (copiadores y copiados), no pudiéndose examinar ninguno de ellos en el presente curso académico (además de cualquiera medida disciplinaria que aplicará el Servicio de Inspección).
Cada tutor organizará una sesión de control de la realización de la práctica:
Es el tutor el que marca la fecha de dicha sesión y no el equipo docente. Los tutores enviarán posteriormente listados de los alumnos que han asistido a dicha sesión.
La asistencia a dicha sesión es obligatoria y se debería realizar antes de la entrega de la práctica en la plataforma aLF.
El tutor puede organizar la sesión hacia el final del cuatrimestre para poder comprobar que los alumnos han hecho bien el trabajo y para ayudar al tutor a calificar el trabajo.
El tutor entrará en el espacio virtual de la asignatura dentro de AGORA antes del 1 de junio, para meter las notas para sus alumnos.

| | |
|---|---|
| Ponderación en la nota final | 1 |
| Fecha aproximada de entrega | 19/05/2025 |
| Comentarios y observaciones | |

**¿CÓMO SE OBTIENE LA NOTA FINAL?**

La ponderación final tanto de la práctica como de cada una de las partes del examen en la nota final de la asignatura quedaría de la siguiente manera:
Práctica Obligatoria: 1 punto sobre 10
Parte Teórica del Examen: 2,5 puntos sobre 10
Parte Práctica del Examen: 6,5 puntos sobre 10

## BIBLIOGRAFÍA BÁSICA

ISBN(13):9788490355312
Título:PROGRAMACION ORIENTADA A OBJETOS CON JAVE USANDO BLUEJ6 edicion
Autor/es:David J. Barnes, Michael Kölling ;
Editorial:PEARSON EDUCACION

Como material complementario, el equipo docente recomienda la consulta de los siguientes documentos:
* Curso OCW de Java de la Universidad Carlos III de Madrid:

httpss://ocw.uc3m.es/course/view.php?id=167

## BIBLIOGRAFÍA COMPLEMENTARIA

## RECURSOS DE APOYO Y WEBGRAFÍA

Como materiales adicionales para el estudio de la asignatura se ofrece en el curso virtual:

* Esta guía de estudio y una versión extendida de la misma, la guía didáctica.
* Enunciados y soluciones de ejercicios teórico-prácticos que el alumno puede usar como ejercicios de autoevaluación.
* Exámenes resueltos de anteriores convocatorias.
* Lista de preguntas frecuentes, que recogen dudas de años anteriores.

Además, a través de CiberUNED se pondrán a disposición de los alumnos herramientas necesarias para el desarrollo de la práctica. Los alumnos pueden acceder al curso virtual en este entorno con su identificador y clave de acceso. En los centros asociados los alumnos dispondrán de ordenadores en donde el entorno de desarrollo BlueJ deberá estar instalado. Además, los alumnos que dispongan de un ordenador personal podrán instalarse dicho entorno de desarrollo. En el entorno CiberUNED se encontrará este paquete software y las instrucciones para su instalación. El entorno virtual se usará como medio para que los estudiantes puedan acceder a material complementario de estudio. El equipo docente añadirá aquellos documentos o referencias que considere adecuados para que los alumnos puedan complementar los conocimientos adquiridos a través del estudio de la bibliografía básica.

## IGUALDAD DE GÉNERO

En coherencia con el valor asumido de la igualdad de género, todas las denominaciones que en esta Guía hacen referencia a órganos de gobierno unipersonales, de representación, o miembros de la comunidad universitaria y se efectúan en género masculino, cuando no se hayan sustituido por términos genéricos, se entenderán hechas indistintamente en género femenino o masculino, según el sexo del titular que los desempeñe.