

GUÍA DOCENTE DE LA ASIGNATURA  
LÓGICA Y PROGRAMACIÓN

Curso: 2019-2020

Fecha última actualización: 18/05/2019

Fecha de aprobación por el Consejo de Departamento: 23/05/2019

MÓDULO	MATERIA	CURSO	SEMESTRE	CRÉDITOS	TIPO
Complementos de Ingeniería del Software	Complementos de Programación	4º	1º	6	Optativa

PROFESOR(ES)	DIRECCIÓN COMPLETA DE CONTACTO PARA TUTORÍAS (Dirección postal, teléfono, correo electrónico, etc.)	HORARIO PARA TUTORÍAS
Fco. Miguel García Olmedo	Departamento de Álgebra, Fac. de Ciencias, planta baja, despacho 2 folmedo@ugr.es <a href="https://www.ugr.es/local/folmedo">https://www.ugr.es/local/folmedo</a>	Consultar en <a href="http://algebra.ugr.es">http://algebra.ugr.es</a> o seguir el código QR: 

GRADO EN EL QUE SE IMPARTE	OTROS GRADOS EN LOS QUE SE PODRÍA OFERTAR
Grado en Ingeniería Informática	

**PRERREQUISITOS Y/O RECOMENDACIONES (Si ha lugar)**

No es necesario que los alumnos tengan aprobadas asignaturas, materias o módulos previos como requisito indispensable para cursar este módulo. No obstante se recomienda la superación de los contenidos y adquisición de competencias de las materias de formación básica y de rama.



UNIVERSIDAD  
DE GRANADA

INFORMACIÓN SOBRE TITULACIONES DE LA UGR  
[grados.ugr.es](http://grados.ugr.es)

Firmado por: LUIS MIGUEL MERINO GONZALEZ Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49 Página: 1 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.

**BREVE DESCRIPCIÓN DE CONTENIDOS (SEGÚN MEMORIA DE VERIFICACIÓN DEL TÍTULO)**

- Programación lógica con Prolog.
- Lambda cálculo.
- Lógica combinatoria.
- Lógica y programación funcional.

**COMPETENCIAS GENERALES Y ESPECÍFICAS****Competencias Generales**

CB4 - Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.

CB5 - Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.

E9 - Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.

**Competencias específicas****Competencias transversales o generales**

T3 - Capacidad de comunicación oral y escrita en el ámbito académico y profesional con especial énfasis en la redacción de documentación técnica.



**UNIVERSIDAD  
DE GRANADA**

**INFORMACIÓN SOBRE TITULACIONES DE LA UGR**  
[grados.ugr.es](http://grados.ugr.es)

Firmado por: LUIS MIGUEL MERINO GONZALEZ Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49 Página: 2 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.

**OBJETIVOS (EXPRESADOS COMO RESULTADOS DE APRENDIZAJE)**

- Conocer el paradigma declarativo de la programación en general y de la programación lógica en particular, como alternativa a otros tipos de programación.
- Conocer los orígenes y la base teórica de la programación lógica. Conocer los aspectos más destacados del lenguaje de programación PROLOG y saber usarlo para la resolución computacional de problemas.
- Conocer y comprender los conceptos de lambda cálculo, los diversos tipos de reducción y su operativa para calcular resultados de funciones.
- Conocer y comprender los conceptos de lógica combinatoria y su relación con el lambda cálculo, como base teórica de la Programación Funcional.
- Conocer el paradigma de la programación funcional como alternativa a otros tipos de programación, y sus diferencias, ventajas e inconvenientes con otros paradigmas.
- Ser capaz de comprender las técnicas más importantes que distinguen a los lenguajes funcionales, identificando la sintaxis y los aspectos básicos de programación en un lenguaje funcional concreto.
- Ser capaz de desarrollar programas en un lenguaje funcional concreto que resuelvan una variedad de problemas tipo.
- Ser capaz de resolver distintos problemas tipo por medio de diferentes paradigmas de programación, funcional, lógico, procedural, usando un intérprete único, y saber comparar las diferentes aproximaciones.
- Ser capaz de 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.



**UNIVERSIDAD  
DE GRANADA**

**INFORMACIÓN SOBRE TITULACIONES DE LA UGR**  
**grados.ugr.es**

Firmado por: LUIS MIGUEL MERINO GONZALEZ Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49 Página: 3 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.

**TEMARIO DE LA ASIGNATURA**

## Temario de Teoría

1. Lógica de Primer Orden: Sintaxis y Semántica.
2. Formas Normales. Unificación y Resolución. Fundamentos de la implementación de Prolog.
3. Notación y la teoría básica: la teoría  $\lambda$ .
4. Reducción: nociones de reducción, el teorema de Church-Rosser, reglas Delta, residuos, formas normales.
5. Lógica Combinatoria: lógica combinatoria, lógica combinatoria y  $\lambda$ -calculus. Algoritmo de abstracción. Supercombinadores.
6. Computabilidad: puntos fijos, sistemas numéricos,  $\lambda$ -expresabilidad.

## Temario de Prácticas

1. Introducción a Prolog: instalación y funcionamiento básicos.
2. Manejo de la sintaxis. Ejemplos de programas paradigmáticos. Manejo de las estructuras de datos primitivos: árboles, listas, conjuntos. Recursos de control y uso del corte. Modificación de la base de conocimiento.
3. Introducción a Haskell: instalación y funcionamiento básicos.
4. Manejo de tipos, constructores, operadores y expresiones primitivas. Clases. Evaluación perezosa.



**UNIVERSIDAD  
DE GRANADA**

INFORMACIÓN SOBRE TITULACIONES DE LA UGR  
[grados.ugr.es](http://grados.ugr.es)

Firmado por: LUIS MIGUEL MERINO GONZALEZ Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49 Página: 4 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.

**BIBLIOGRAFÍA****BIBLIOGRAFÍA FUNDAMENTAL**

- Barendregt, H.P. The Lambda Calculus. Its Syntax and Semantics. North-Holland, 1984.
- Clocksin, W.F.; Mellish, C.S. Programming in Prolog, 5th Edition. Springer Verlag.
- Chin-Liang Chang; Chart-Tung Lee, R. Symbolic Logic and Mechanical Theorem Proving. Academic Press, 1973.
- Delahaye, J.P. Formal Methods in Artificial Intelligence. Willey, 1987.
- Hindley, J.R.; Seldin, J.P. Lambda-Calculus and Combinators. And Introduction. Cambridge University Press, 2008.
- Ruiz, B.C., Gutiérrez, F. y otros. Razonando con Haskell. Thomson, 2004.

**BIBLIOGRAFÍA COMPLEMENTARIA**

- Bird, R. Introducción a la Programación Funcional con Haskell. Prentice Hall, 2000.
- Hortalá T.; Martí, N. y otros. Lógica Matemática para Informáticos. Ejercicios Resueltos. Prentice Hall Pearson, 2008.
- Lipovaca, M. Learn you a Haskell for Great Good! A beginner's Guide. No Starch Press, 2011.
- Mendelson, E. Introduction to mathematical logic, Chapman and Hall, 1997.
- Sterling L.; Shapiro, E. The Art of Prolog: Advanced Programming Techniques. MIT Press, 1986.



**UNIVERSIDAD  
DE GRANADA**

**INFORMACIÓN SOBRE TITULACIONES DE LA UGR**  
**grados.ugr.es**

Firmado por: LUIS MIGUEL MERINO GONZALEZ Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49 Página: 5 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.

**ENLACES RECOMENDADOS**

- [Página oficial de Haskell](#)
- [Aprende Haskell](#)
- [Real World Haskell](#)
- [Manual de Prolog](#)
- [Learn prolog now](#)
- [IHaskell](#)
- [Calysto prolog](#), un núcleo de prolog para [jupyter](#)
- Natural language toolkit [nltk](#)

**ACTIVIDADES FORMATIVAS**

Lección Magistral (Clases Teóricas-Expositivas):

Actividades Prácticas (Clases prácticas en Laboratorio):

Actividades no presenciales Individuales.

Actividades no presenciales grupales.

Tutorías académicas.



**UNIVERSIDAD  
DE GRANADA**

**INFORMACIÓN SOBRE TITULACIONES DE LA UGR**  
[grados.ugr.es](https://grados.ugr.es)

Firmado por: LUIS MIGUEL MERINO GONZALEZ Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49 Página: 6 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.

**METODOLOGÍA DOCENTE**

Lección Magistral: Presentación en el aula de los conceptos propios de la materia haciendo uso de metodología expositiva con lecciones magistrales participativas ilustradas con variedad de ejemplos.

Resolución de problemas: Realización en el aula de distintos ejercicios donde se abordan los diferentes tópicos de la asignatura.

Prácticas en Laboratorio: En el laboratorio probaremos diferentes formas de resolver problemas y analizaremos la eficiencia de éstas.

Exposición de Trabajos Tutelados: Los estudiantes deberán realizar un trabajo individualmente o en grupos (de hasta 3 personas) sobre un tema elegido por ellos. Este trabajo deberán exponerlo ante los demás estudiantes, que podrán preguntar y debatir sobre el tema.

Tutorías académicas: En el horario establecido, los estudiantes podrán realizar consultas personalizadas al profesor, así como exponer los distintos trabajos prácticos que hayan realizado.



**UNIVERSIDAD  
DE GRANADA**

INFORMACIÓN SOBRE TITULACIONES DE LA UGR  
[grados.ugr.es](http://grados.ugr.es)

Firmado por: LUIS MIGUEL MERINO GONZALEZ Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49 Página: 7 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.

**EVALUACIÓN (INSTRUMENTOS DE EVALUACIÓN, CRITERIOS DE EVALUACIÓN Y PORCENTAJE SOBRE LA CALIFICACIÓN FINAL, ETC.)**

Todo lo relativo a la evaluación se registrará por la Normativa de evaluación y calificación de los estudiantes vigente en la Universidad de Granada, que puede consultarse en **Secretaría General**. Preferentemente, la evaluación se ajustará al sistema de evaluación continua del aprendizaje del estudiante siguiendo el artículo 7 de la anterior Normativa.

El criterio de evaluación se especifica a continuación:

- Un 50 % de la evaluación se basará en las prácticas entregadas y defendidas por los alumnos dentro de los plazos que se fijarán a lo largo del curso.
- Un 30 % de la evaluación se basará en la elaboración y presentación ante el profesor y el resto de estudiantes de un trabajo sobre un tema elegido por el estudiante.
- Un 20 % de la nota vendrá dada por el examen teórico práctico que se realizará una vez finalizado el curso.

Para los estudiantes que se acojan a la evaluación única final, esta modalidad de evaluación estará formada por todas aquellas pruebas que el profesor estime oportunas, de forma que se pueda acreditar que el estudiante ha adquirido la totalidad de las competencias generales y específicas descritas en el apartado correspondiente de esta Guía Docente.

En las convocatorias extraordinarias la evaluación consistirá en un examen general.

El sistema de calificaciones se expresará mediante calificación numérica de acuerdo con lo establecido en el art. 5 del R. D 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de calificaciones en las titulaciones universitarias de carácter oficial y validez en el territorio nacional.

**RÉGIMEN DE ASISTENCIA**

- La asistencia a las clases teóricas no será obligatoria, aunque la participación activa en clase y la entrega de ejercicios planteados por el profesor se tendrá en cuenta dentro del sistema de evaluación continua de la asignatura.
- La asistencia a las clases prácticas no será obligatoria, exceptuando las sesiones en las que se programen pruebas de evaluación. En cualquier caso, la asistencia y participación activa en clase se tendrá en cuenta dentro del sistema de evaluación continua de la asignatura.



**UNIVERSIDAD  
DE GRANADA**

**INFORMACIÓN SOBRE TITULACIONES DE LA UGR**  
**grados.ugr.es**

Firmado por: LUIS MIGUEL MERINO GONZALEZ Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49 Página: 8 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.

**DESCRIPCIÓN DE LAS PRUEBAS QUE FORMARÁN PARTE DE LA EVALUACIÓN ÚNICA FINAL ESTABLECIDA EN LA “NORMATIVA DE EVALUACIÓN Y DE CALIFICACIÓN DE LOS ESTUDIANTES DE LA UNIVERSIDAD DE GRANADA”**

Según la normativa vigente, la evaluación única final, entendiéndose por tal la que se realiza en un solo acto académico, podrá incluir cuantas pruebas sean necesarias para acreditar que el estudiante ha adquirido la totalidad de las competencias descritas en la Guía Docente de la asignatura.

**UNIVERSIDAD  
DE GRANADA****INFORMACIÓN SOBRE TITULACIONES DE LA UGR**  
**grados.ugr.es**

Firmado por: LUIS MIGUEL MERINO GONZALEZ    Director/a de Departamento

Sello de tiempo: 23/05/2019 11:19:49    Página: 9 / 9



eK65BSq08n+mQ+74yAz1wH5CKCJ3NmbA

La integridad de este documento se puede verificar en la dirección <https://sede.ugr.es/verifirma/pfinicio.jsp> introduciendo el código de verificación que aparece debajo del código de barras.