

Departamento de Álgebra. Curso 2010/2011

Guía Docente de Asignaturas

NOMBRE: Lógica y Métodos Discretos

TITULACIÓN: Grado en Ingeniería Informática

CURSO: 1

NÚM. DE CRÉDITOS: Total: 6

Teoría: 4.5

Prácticas: 1.5

CARÁCTER: (Anual: Primer C.: Segundo C.:
(Básica: Obligatoria: Optativa:)

REDACTOR: Francisco Miguel García Olmedo

CONTACTO: folmedo@ugr.es

DESCRIPCIÓN DE LA ASIGNATURA:

1. Listado de descriptores:

- a) Inducción y recurrencia.
- b) Conjuntos Ordenados. Técnicas recursivas. Retículos y Álgebras de Boole. Álgebra Combinacional.
- c) Métodos en estructuras discretas.
- d) Lógica Proposicional. Aplicaciones a la programación.
- e) Lógica de Predicados. Representación lógica del conocimiento.
- f) Programación Lógica: Unificación y Resolución.

2. Recomendaciones y conocimientos previos:

- a) Son prerequisites: familiaridad con el lenguaje matemático básico y determinados paquetes de cálculo, como Maxima.
- b) Es recomendable: haber superado las asignaturas del grado, primer cuatrimestre, con contenido matemático.

3. Objetivos de la asignatura:

- a) Conocimiento del concepto de conjunto ordenado y reconocimiento de los elementos distinguidos en un conjunto ordenado.
- b) Conocimiento de ejemplos de órdenes que sean de utilidad, en especial el orden lexicográfico y el lexicográfico inverso.
- c) Conocimiento y comprensión de el principio de inducción y aplicarlo para probar fórmulas sencillas así como para construir funciones recursivas.
- d) Capacidad para resolver relaciones de recurrencia sencillas: lineales de primer orden y lineales de segundo orden con coeficientes constantes.

- e) Capacidad para entender los retículos como ejemplos de conjuntos ordenados y obtener las álgebras de Boole como ejemplos de éstos.
- f) Conocimiento de la estructura de las álgebras de Boole finitas.
- g) Capacidad para la construcción de ejemplos de álgebras de Boole finitas, como las de funciones booleanas.
- h) Capacidad para aplicar métodos para la minimización de sistemas combinacionales, como el de los mapas de Karnaugh y el algoritmo de Quine-McCluskey.
- i) Capacidad para describir el lenguaje proposicional y el concepto de interpretación de una fórmula bien formada.
- j) Capacidad para expresar conectivas habituales en la programación procedural en términos de proposiciones, y aplicar la lógica proposicional a la programación informática.
- k) Capacidad para comprender los conceptos de tautología, satisfacible/insatisfacible y contradicción para una fórmula.
- l) Capacidad para utilizar herramientas como las tablas de verdad, las interpretaciones semánticas y otras para reconocer el carácter de una fórmula.
- m) Capacidad para la utilización de software simbólico en el cálculo del carácter de una fórmula, su interpretación, su dual y otros conceptos relacionados.
- n) Capacidad para transformar problemas de consecuencia lógica en problemas de inconsistencia de un conjunto de cláusulas y resolverlos mediante el uso de diversas técnicas.
- ñ) Comprensión de la lógica de predicados y el concepto de interpretación.
- o) Capacidad para obtener la Forma Clausular de una fórmula.
- p) Capacidad para aplicar el método de resolución con unificación para determinar el carácter de inconsistencia de un conjunto de cláusulas.
- q) Conocimiento y capacidad de uso de algunos criterios de búsqueda de la inconsistencia usando resolución, en especial el de resolución lineal ordenada, con sus aplicaciones en el lenguaje de programación lógica PROLOG.
- r) Conocimiento y comprensión del método de Resolución PROLOG, y de cómo este es un caso especial de resolución lineal ordenada.
- s) Conocimiento básico de la Sintaxis PROLOG, y su aplicación práctica mediante software a problemas vistos de resolución, para la obtención de las soluciones.
- t) Capacidad de utilización de PROLOG para resolver problemas de Unificación.
- u) Capacidad de programación básica en PROLOG: modelización de problemas básicos expresables mediante reglas y hechos en PROLOG, así como de uso del mismo para su solución.

4. Estructura (en horas de trabajo del estudiante):

Total Presencial: 60

a) Número de horas de clases de teoría: (20 teoría + 10 problemas=) 30

- b) Número de horas de clases de problemas: 15
- c) Número de horas de clases prácticas: (5 teoría + 10 problemas=)15
- d) Número de horas de participación en seminarios, exposiciones, trabajo en grupos reducidos, ...

Total no presencial: 0

- e) Preparación de trabajos académicamente dirigidos y otras actividades:
- f) Estudios de clases presenciales:

Total: 60

TÉCNICAS DOCENTES (Metodología):

1. Técnicas docentes utilizadas:

- a) Clases teóricas. Durante estas sesiones el profesor expondrá los elementos más significativos del tema, proporcionará ejemplos y planteará, en algunos casos, ejercicios que permitan al alumno familiarizarse con aspectos de la materia que presenten dificultades de manipulación.
- b) Clases prácticas. Durante estas sesiones se desarrollarán ejemplos paradigmáticos de aplicación de la teoría, con o sin ayuda de computadora, y se ejemplificarán las técnicas más representativas de trabajo en la materia. Estas clases serán necesariamente participativas.
- c) Exposiciones y seminarios. Los alumnos, organizados en grupos, deberán desarrollar un tema elegido de mutuo acuerdo con el profesor y exponerlo en presencia de sus compañeros.
- d) Conferencias. El profesor solicitará la asistencia a conferencias específicas sobre la materia que se desarrollen durante el periodo de docencia.

2. Desarrollo del curso y justificación:

Cada semana se impartirán 4 horas de clase en las que se desarrollarán los aspectos teóricos y procedimentales de la asignatura. Las 4 horas se repartirán en: 3 de teoría y problemas que versarán sobre los contenidos de la asignatura y 1 de laboratorio, con teoría y problemas, cuya resolución será basada en el uso de la computadora y del software libre.

PROGRAMA DE LA ASIGNATURA:

1. Inducción y recurrencia.

- a) Inducción.
- b) La relación de recurrencia lineal de primer orden.
- c) La relación de recurrencia homogénea lineal de segundo orden con coeficientes constantes.
- d) La relación de recurrencia no homogénea.

2. Retículos y álgebra de Boole:

- a)* Conjuntos ordenados.
- b)* Retículos.
- c)* Retículos Distributivos.
- d)* Retículos Complementados.
- e)* Álgebra de Boole.
- f)* Teoremas fundamentales del Álgebra de Boole.
- g)* Representación atómica de las Álgebras de Boole finitas.
- h)* Expresiones Booleanas.
- i)* Circuitos combinacionales.
- j)* Formas normales de expresiones booleanas.
- k)* Método de Quine-McCluskey.

3. Combinatoria.

- a)* Métodos elementales de conteo: principio de inclusión–exclusión, principio del producto, variaciones, el principio del palomar.
- b)* Combinaciones.
- c)* Permutaciones.

4. Introducción a la teoría de grafos.

- a)* Matrices asociadas a grafos.
- b)* Isomorfismo de grafos.
- c)* Grafos de Euler.
- d)* Grafos de Hamilton.
- e)* Grafos bipartidos.
- f)* Grafos planos.
- g)* Coloración de grafos.
- h)* Árboles.

5. Lógica Proposicional.

- a)* Lenguaje Proposicional.
- b)* Implicación semántica. Propiedades básicas.
- c)* Implicación semántica y satisfacibilidad.
- d)* Equivalencia lógica: caracterizaciones y propiedades.
- e)* Equivalencia lógica como congruencia: Álgebra de Lindenbaum-Tarski, el álgebra de Boole de las proposiciones lógicas.
- f)* Forma normal conjuntiva: método sintáctico versus método de las tablas de verdad.

- g) Resolución sin variables. Implementación: Método de Davis y Putnam.
6. Lógica de Predicados. Representación del Conocimiento.
- a) Lenguajes de primer orden.
 - b) Interpretación, satisfacibilidad y verdad.
 - c) Forma normal prenexa y de Skolem.
 - d) Teorema de Herbrand.
7. Programación Lógica: Unificación y Resolución.
- a) Algoritmo de Unificación.
 - b) Principio de Resolución.
 - c) Regla de resolución y su administración: gestión de conjuntos de cláusulas y exploración del árbol de las deducciones.
 - d) Exploración del árbol de las deducciones: primero en profundidad y primero en anchura.
 - e) Exploración de subárboles: estrategias lineales, estrategia Input, estrategia Unit, estrategias ordenadas.
 - f) Introducción al lenguaje Prolog: resolución Prolog, control, aritmética y tratamiento de listas. Ejemplos.

PROGRAMACIÓN DE PRÁCTICAS:

1. Inducción.
2. Recurrencia.
3. Retículos y Álgebras de Boole.
4. Funciones booleanas.
5. Grafos.
6. Combinatoria.
7. Lógica proposicional (I).
8. Lógica proposicional (II).
9. Introducción a prolog (sintaxis).
10. y siguientes. Prolog (resolución y unificación, recursividad, elementos de programación).

BIBLIOGRAFÍA:

1. BIGGS, N.L. *Matemática Discreta*. Vicens Vives, 1994.

2. CHANG C. and LEE, R.C. *Symbolic Logic and Mechanical Theorem Proving*. Academic Press, 1973.
3. DELAHAYE, J.P. *Outils Logiques pour l'Intelligence Artificielle*. Eyrolles, 1986.
4. DEO, N. *Graph Theory with Applications to Engineering and Computer Science*. Prentice-Hall, 1974.
5. GRIMALDI, R.P. *Matemática Discreta y Combinatoria*. Addison-Wesley Publishing Company, 1998.
6. HILL, F.J. and PETERSON, G.R. *Teoría de Commutación y Diseño Lógico*. Limusa, 1993.
7. LIPSCHUTZ, S. and LIPSON, M. *2000 problemas resueltos de Matemática Discreta*. McGraw Hill, 1998.
8. PERMINGEAT, N. and GLAUDE, D. *Álgebra de Boole: Teoría, Métodos de Cálculo y Aplicaciones*. Vicens Vives, 1995.
9. ROSEN, K. H. *Matemática Discreta y sus Aplicaciones*. McGraw Hill, 2003.
10. STERLING, L. and SHAPIRO, E. . *The Art of Prolog : advanced programming techniques*. MIT Press, 1975.
11. VEERARAJAN, T. *Matemática Discreta*. McGraw Hill, 2008.
12. YABLONSKY, S.V. *Introduction to Discrete Mathematics*. Mir, 1975.

DESTREZAS QUE SE PRETENDE QUE EL ALUMNO ALCANCE TRAS SEGUIR LA ASIGNATURA:

GENERALES.

1. Desarrollar la capacidad de análisis y síntesis, el razonamiento crítico, el aprendizaje autónomo, el trabajo en equipo, la iniciativa y el espíritu emprendedor.-
2. Ejercitar en la resolución de problema.-
3. Ejercitar la capacidad de usar paquetes de cálculo y programar.-

ESPECÍFICAS.

1. Saber emplear la técnica de inducción para resolver problemas.
2. Saber plantear relaciones de recurrencia para resolver problemas.
3. Entender la relación entre la axiomática del Álgebra de Boole y su orden subyacente.
4. Conocer la estructura del álgebra de Boole finita.
5. Conocer la relación entre el álgebra de Boole y los circuitos combinatoriales.
6. Saber simplificar la expresión de un circuito usando las técnicas del álgebra de Boole.

7. Conocer y saber usar los principales métodos de conteo.
8. Conocer los principales resultados sobre grafos y saber modelar problemas a través del lenguaje de los grafos.
9. Conocer la estructura de árbol y saber usarla para resolver problemas.
10. Conocer el concepto de fórmula proposicional, el de implicación semántica y el de equivalencia lógica.
11. Conocer y saber encontrar las formas normales de una fórmula dada.
12. Conocer y saber emplear el método de Davis y Putnam como herramienta óptima para solucionar problemas de insatisfacibilidad e implicación semántica.
13. Conocer el lenguaje de primer orden y las fórmulas de dicho lenguaje.
14. Saber encontrar las formas normales de fórmulas de primer orden.
15. Conocer y saber emplear el algoritmo de unificación y los métodos de resolución más eficaces.
16. Conocer los rudimentos de Prolog.

EVALUACIÓN: Para la evaluación se considerarán los siguientes elementos:

1. Resultado de las pruebas escritas.
2. Participación en clases prácticas.
3. Participación en los seminarios.
4. Participación en las tutorías colectivas.
5. Asistencia a tutorías individuales.
6. Asistencia a clases teóricas y prácticas.
7. Entrega de tareas que sean solicitadas por el profesor.

Durante el curso 2010-2011 la nota final obtenida por el alumno en la evaluación de la asignatura se calculará sobre un máximo de 10 puntos de la siguiente forma: Hasta 2 puntos la evaluación de las prácticas, de los cuales 0,7 puntos se obtendrán con la asistencia a, al menos, el 75 % de las sesiones prácticas, y el resto será el resultado de una prueba específica. Hasta 8 puntos en la evaluación de la parte teórica y de problemas.

ENLACES DE INTERÉS:

1. http://johnlapeyre.com/qinf/qinf_html/maxima_html/maxima_toc.html
2. http://www.delorie.com/gnu/docs/maxima/maxima_toc.html

3. <http://webs.um.es/mira/maxima/manualico.html>

4. <http://webs.um.es/mira/maxima/Matrices.html>

5. <http://www.ugr.es/local/folmedo>