

**NOMBRE DE LA ASIGNATURA:**

**PROGRAMACIÓN LÓGICA**

**Año 2025**

**Carrera/ Plan: (Dejar lo que corresponda)**

*Licenciatura en Informática Plan 2021/Plan 2015*

**Año: 2025**

**Régimen de Cursada: Semestral**

**Carácter (Obligatoria/Optativa): Optativa**

**Correlativas: Conceptos y Paradigmas de Lenguajes de Programación**

**Profesor/es: Smith, Clara**

**Hs. semanales teoría: 3 (tres)**

**Hs. semanales práctica: 6 (seis)**

## **FUNDAMENTACIÓN**

Conocer los fundamentos de la Programación Lógica incrementa en los alumnos la capacidad de razonamiento abstracto y los prepara para abordar las actuales y complejas cuestiones ligadas al manejo y representación computacional de conocimiento, especialmente las teorías actuales de **agentes inteligentes** y de aprendizaje automático (**machine learning**).

Para ello, el curso trata los siguientes temas: la lógica como lenguaje de representación de conocimiento, fundamentos de la programación lógica, lógica modal, diseño de sistemas multiagentes, e inferencias lógicas para machine learning. Prolog, el lenguaje paradigmático de la programación declarativa, es usado para la programación de algunos módulos básicos.

## **OBJETIVOS GENERALES**

Conocer los **fundamentos teóricos y prácticos** de la Programación Lógica (**por qué “anda”**).

Manejar y prever algunos aspectos computacionales de estos lenguajes y de combinaciones de ellos (como la **semántica** y la **decidibilidad** de las lógicas subyacentes).

Adquirir la habilidad de **diseñar sistemas de agentes**.

Comprender **el algoritmo de machine learning** que está en la **base de todos los modernos algoritmos de aprendizaje simbólico** computacional.

**Fortalecimiento en la base innovativa** desde la perspectiva de la **ingeniería de software**, al usar diferentes lenguajes lógicos como lenguajes de representación de conocimiento.

## **RESULTADOS DE APRENDIZAJE**

Al finalizar el curso, se espera que los resultados de aprendizaje sean:

- 1- Un incremento en la capacidad de **razonamiento abstracto**.
- 2- La obtención de una nueva base cognitiva para abordar el **manejo y la representación computacional** de conocimiento especialmente ligado a **agentes inteligentes**.
- 3- El conocimiento de los tipos de inferencias en algoritmos de **machine learning**.

## **COMPETENCIAS**

LI-CE2- Planificar, dirigir, realizar y/o evaluar proyectos de especificación, diseño, verificación, validación, puesta a punto, mantenimiento y actualización para redes de comunicaciones que vinculen sistemas de procesamiento de datos. Esto incluye comunicaciones convergentes y unificadas, así como redes definidas por software y redes virtuales. En particular, desarrollar las soluciones de las capas superiores de los protocolos de red, a partir del hardware que se haya seleccionado.

LI-CE4- (i) Planificar, dirigir, realizar y/o evaluar proyectos de relevamiento de problemas del mundo real. (ii) Especificación formal de los mismos, diseño implementación, prueba, verificación, validación, mantenimiento y control de calidad de sistemas de software/sistemas de información que se ejecuten sobre equipos de procesamiento de datos con capacidad de incorporación de tecnologías emergentes del cambio tecnológico. (iii) Capacidad de análisis, diseño y evaluación de interfases humano computador y computador-computador.

### **CONTENIDOS MÍNIMOS (de acuerdo al Plan de Estudios)**

- Conceptos de Inteligencia Artificial.
- Inteligencia artificial simbólica y no simbólica.
- Lógica matemática y lógicas aplicadas.
- El paradigma declarativo.
- Lenguajes de programación lógica (Prolog).

### **PROGRAMA ANALÍTICO**

#### **1. Fundamentos.**

La Lógica de Primer Orden (LO1) como lenguaje de representación del conocimiento. La fórmula bien formada de la LO1 vista como programa. Indecidibilidad de la LO1. Cláusulas de Horn. Notación clausal: Algoritmo.

#### **2. Enfoque orientado a modelos.**

Interpretación para un conjunto de cláusulas. Modelo para un conjunto de cláusulas. Teoría de Herbrand: universo de Herbrand, base de Herbrand, interpretación de Herbrand, modelo de Herbrand. Interpretación de Herbrand asociada a una interpretación. Conjunto de partes de la Base de Herbrand. La organización de interpretaciones de Herbrand como un reticulado. Instancia básica de una cláusula. Teorema de Herbrand. La propiedad de intersección de modelos. Semántica declarativa de un programa lógico.

#### **3. Enfoque orientado a la demostración automática de teoremas.**

Sustitución: definición y propiedades. Unificación: unificador más general y algoritmo de unificación. Regla de Resolución. Resolvente bajo una sustitución.

#### **4. Estrategias de Resolución.**

Regla de Resolución para cláusulas de Horn. Saturación. Espacio de resolventes. Búsqueda en el espacio de resolventes. Refinamiento de métodos. El filtrado de tautologías. El filtrado de literales puros. Técnicas de simplificación: subsunción y factorización. El conjunto soporte. Conjunto inicial. Resolución lineal. Aplicaciones combinadas de métodos. SLD-refutación como refinamiento de resolución lineal. Árbol-SLD: tipos de ramas. Teorema de Hill. Teorema de Clark. Relación entre semántica declarativa y semántica procedural de un programa lógico. Estrategia de selección de átomos, propiedad de independencia de la regla de computación. Orden de las cláusulas. Estrategia de recorrido del árbol. Funcionamiento de un sistema Prolog. Completitud: instancias en la que puede quedar comprometida. Uso del cut (!) en un programa lógico.

#### **5. Razonamiento no monótono.**

El conocimiento "positivo". Problemas que necesitan de conocimiento "negativo". Hipótesis de mundo cerrado (CWA). Caracterización "a la Herbrand". Hipótesis de clausura de dominio (DCA) y su relación con la teoría de Herbrand. Suposición de nombres únicos (UNA) y su vinculación con la regla de paramodulación. No-monotonía. CWA para cláusulas generales. Negación por falla: definición, su relación con CWA. Árbol-SLD finitamente fallado.

#### **6. Lógica Modal y Sistemas Multiagentes.**

Estructuras relacionales. Operadores modales de necesidad y posibilidad. Significado intuitivo. Semántica formal: la semántica de Kripke de mundos posibles. Frames y Modelos. Ejemplos de lógicas modales: dinámica, temporal, deóntica. Sus operadores y axiomatizaciones. **Grupos y sistemas multiagentes.** Combinación de lógicas: definición. Aspectos vinculados a la ingeniería de software: modularidad. Transferencia de propiedades de las lógicas especiales a las lógicas combinadas: completitud y decidibilidad.

#### **7. Aprendizaje Simbólico Computacional.**

Inferencia de patrones. Razonamiento inductivo y razonamiento deductivo. Algoritmo de Mitchell: conjuntos frontera. Conjuntos de instancias de entrenamiento. Instancias positivas y negativas. Sesgo conceptual. Esquemas de inferencia para el **aprendizaje de patrones** a partir de instancias.

## **BIBLIOGRAFÍA**

- P. Blackburn, M. de Rijke, Y. Venema. Modal Logic. Cambridge University Press, 2001.
- H. G. Hamilton. Lógica para Matemáticos. Paraninfo, Madrid, 1981.
- J. W. Lloyd. Foundations of Logic Programming. Second ed. Springer Verlag, 1993.

Generalization as search. Mitchell.

## **METODOLOGÍA DE ENSEÑANZA**

Las experiencias en 2020 y 2021 en la modalidad virtual; y semipresencial en 2022, 2023 y 2024 fueron exitosas por ello se prevé continuar la semipresencialidad en 2025. Las horas totales estimadas para el curso son 96 hs., no obstante está previsto un máximo de duración de 16 semanas.

El alumno asiste a clases presenciales a convenir con la cátedra y a reuniones preestablecidas vía teleconferencia de ser requeridas. La evaluación del curso es presencial en su formato.

Al ser ésta una asignatura optativa, normalmente el número de alumnos que cursa Programación Lógica es de sencilla administración por parte del grupo docente. Esto sucedía ya previo a la cuarentena, pues muchos alumnos (si no todos) al llegar a cursarla ya están hacia el final de su carrera, tienen organizados sus trabajos/esquemas laborales, y la flexibilidad es una prioridad para ellos, y también lo es para la cátedra en un intento por acompañarlos en la finalización de todos sus cursos.

Antecedentes. El contenido de este programa fue impartido de menor a mayor intensidad, ininterrumpidamente desde 2004 hasta la fecha, en las asignaturas Inteligencia Artificial Plan 90, Lógica, Programación Lógica, Lógica y Elementos de IA y Lógica e IA. El 2014 fue el primer año en el que algunos temas fueron también presentados a alumnos con formación muy básica (3er. año): ocho alumnos aprobaron en 2014 el examen, dos con nota buena, uno con nota distinguida y cinco con nota suficiente. Ello ha aportado a una transversalidad con los contenidos de la asignatura "Lógica". Asimismo, parte de los contenidos, especialmente los referidos a Lógica Modal, colaboran con la transversalidad en la materia "Teoría de la Computación". Finalmente cabe destacar que tres tesis de grado en la temática han sido aprobadas exitosamente en nuestra Facultad, con trabajos presentados en congresos (un trabajo con premio nacional en 2011). Uno de los tesisistas fue además mejor promedio de la Facultad en su cohorte. Otro tesisista obtuvo en 2017 su doctorado en la U. de Luxemburgo en esta temática, y otro tesisista fue durante 5 años becario doctoral de Conicet. La cátedra elaboró un libro de Notas de Cátedra que se publicó por Editorial UNLP en 2024. Dicho texto está disponible para los alumnos y el público en general en <https://sedici.unlp.edu.ar/handle/10915/162904>. Alumnos de las promociones 2021 y 2023 colaboraron como correctores de dicho texto de cátedra.

Organización de las clases. Las clases son teórico-prácticas, organizadas por bloques que se corresponden cada uno con una bolilla del programa. Al inicio del bloque se dictan la o las clases referidas a la bolilla en cuestión. En las clases subsiguientes los alumnos realizan la exposición de diferentes ejercicios teóricos o prácticos que les han sido asignados para su resolución. Así, la corrección es individual y también grupal, ya que al ser una exposición oral el auditorio realiza los comentarios pertinentes a la exposición. Este sistema ha dado buenos resultados en los últimos años, debido a que el número de inscriptos de cada cohorte vuelve a los grupos apto para la comunicación cercana y la exposición oral.

Aspecto exitoso de las clases de Programación Lógica desde las últimas cohortes, en relación con la transversalidad de la asignatura con otras materias. Algunos alumnos al cursar la asignatura Lógica e IA (asignatura en la que docentes de Programación Lógica también participan) descubren el interés por la asignatura y traen un bagaje de conocimiento importante que les permite insertarse fácilmente en este

curso y finalizarlo exitosamente.

Contenidos del programa de Programación Lógica. Aparecen en un orden secuencial tal que cada tema constituye un bloque básico para el abordaje y estudio de los temas que le siguen. Retomamos los aspectos básicos de la lógica proposicional, y seguidamente incursionamos en la Lógica de Predicados. Luego vemos las dificultades de la lógica proposicional para capturar discursos sofisticados tales como el normativo (obligatoriedad/permiso) o el temporal (en el futuro sucede/en el pasado sucedió). Luego ampliamos la visión para definir formalmente sistemas donde intervienen múltiples agentes. Los principios básicos que rigen a los sistemas multiagentes son caracterizables a partir de atribuciones internas de los agentes -y grupos de agentes- tales como capacidad, intención, creencias, preferencias; y otras actitudes como persistencia y efectividad en el accionar.

Para cada cohorte se define además un grupo virtual. A ese espacio compartido se suben archivos relevantes, y en él se discuten cuestiones referidas a los temas teóricos y a los prácticos.

## EVALUACIÓN

### Mecanismos de seguimiento y evaluación durante el curso.

Se realizan exposiciones individuales por bolilla (resolución de uno o más ejercicios por práctica).

Seguimiento: las diferentes exposiciones son individuales. Por el tipo de contenido de la asignatura, cada exposición permite evaluar el afianzamiento en el alumno de los temas de las bolillas anteriores a la que se está exponiendo. Cada exposición lleva una calificación numérica.

### Condiciones para la aprobación de los trabajos prácticos.

La aprobación de los trabajos prácticos se obtiene con la aprobación de todas las exposiciones.

Clases específicas y temas especiales se definen únicamente si el alumno, luego de aprobar lo requerido, decide mejorar su calificación. Con ello se aspira a que el plus de nota final que pudiese obtener sea reflejo de la profundización de algún tema de la asignatura de particular interés del alumno.

### Pautas para la aprobación del examen final para los alumnos que han aprobado los trabajos prácticos.

Todas las exposiciones individuales (resolución de uno o más ejercicios por práctica). Pueden sumarse exposiciones extra a requerimiento del alumno para lograr un desempeño destacado.

### Mecanismos para la promoción del examen final:

Todas las exposiciones individuales.

### Forma de evaluar competencias y resultados de aprendizaje.

Las competencias y resultados del aprendizaje se evalúan a partir de las exposiciones en el pizarrón, sumado al hecho que, de modo casi inmediato al comienzo del curso, los alumnos adquieren el vocabulario técnico del área, y ganan en precisión al expresarse oralmente y de modo escrito. El uso adecuado del vocabulario técnico es, de por sí, un indicador de la aprehensión de los conceptos del área. También la capacidad de búsqueda de material específico por sí mismos denota en los alumnos la evolución en la internalización de los temas tratados, altamente especializados. La capacidad para elegir el tema de tesis de grado y de cursos a tomar, una vez finalizado este curso.

## **CRONOGRAMA DE CLASES Y EVALUACIONES**

Clase	Fecha	Contenidos/Actividades
1	24 ago	Introducción. SAT. FNC.Decidibilidad. FNC para LO1. Representación clausal.
2	31 ago	Satisfactibilidad de la FNC. Compacidad. Resolución. Teorema de Robinson.
3	7 sept	Cláusulas generales y de Horn. Semántica declarativa y procedural de los programas lógicos. Teoría de Herbrand.



4	14 sept	Sustitución. Unificación y Resolución
5	28 sept	Estrategias de Resolución SLD-refutación, Correctitud y Completitud de SLD-refutación, Aspecto procedural de programas lógicos.
6	5 oct	CWA y Negación por falla.
7	19 oct	Prolog. Aspectos declarativos.
8	26 oct	Prolog. Aspectos procedurales.
9	2 nov	Programas en Prolog.
10	9 nov	La lógica modal como KRL (lenguaje de representación de conocimiento).
11	16 nov	Lógica modal: sintaxis y semántica. Extensiones del cálculo de enunciados.
12	23 nov	Ejemplos de lógicas modales. Teorías de Agentes.
13	30 nov	Sistemas de agentes.
14	7 dic	Machine Learning. Conceptos.
15	14 dic	Algoritmo de Mitchell
16	21 dic	Cierre

Evaluaciones previstas	Fecha
LO1 como KRL	28 sept
Teoría de Herbrand y Resolución	19 oct
Prolog	16 nov
lógica modal y machine learning	14 dic

**Contactos de la cátedra:**

- **Mail (obligatorio):** [claritasmith@gmail.com](mailto:claritasmith@gmail.com)
- **Sitio WEB:**
- **Plataforma virtual:**
- **Otros:** <https://sedici.unlp.edu.ar/handle/10915/162904>

Firma del/los profesor/es