

# Clase de Inteligencia Artificial: Proyecto de Representación del Conocimiento

## Profesores:

Luis A. Pineda Cortés, IIMAS, UNAM  
Arturo Rodríguez García, PCIC, UNAM

Septiembre 17, 2015

## Condiciones de Entrega

**Formato:** Código fuente dentro de una carpeta comprimida llamada *Proyecto1.zip*. Archivo *doc.pdf* con la documentación del proyecto que incluya la descripción de los predicados solicitados en la descripción del proyecto (indicando a qué se refieren sus argumentos y mostrando por lo menos dos ejemplos de uso).

**Fecha y modo de entrega:** Enviar el código y la documentación por correo electrónico a más tardar el 13 de octubre a las 11:00 am. Enviar correo con asunto *Proyecto de Representación del Conocimiento* y archivos adjuntos a [lpineda@unam.mx](mailto:lpineda@unam.mx) y [arturo8602@yahoo.com.mx](mailto:arturo8602@yahoo.com.mx). Entregar un CD con el código fuente y una copia impresa de la documentación justo al inicio de la clase del 13 de octubre.

## Descripción

A partir de la jerarquía conceptual con defaults y excepciones explicada en clase y de la especificación de la base de conocimiento presentada en el artículo “A Light Non-Monotonic Knowledge-Base for Service Robots”:

1. Crear predicados para consultar:
  - a) La extensión de una clase (el conjunto de todos los objetos que pertenecen a la misma, ya sea porque se declaren directamente o porque están en la cerradura de la relación de herencia).
  - b) La extensión de una propiedad (mostrar todos los objetos que tienen una propiedad específica ya sea por declaración directa o por herencia, incluyendo su respectivo valor).
  - c) La extensión de una relación (mostrar todos los objetos que tienen una relación específica ya sea por declaración directa o por herencia, incluyendo con quién están relacionados).
  - d) Todas las clases a las que pertenece un objeto
  - e) Todas las propiedades de un objeto o clase
  - f) Todas las relaciones de un objeto o clase
2. Crear predicados para añadir:

- a) Clases u objetos
  - b) Propiedades nuevas a clases u objetos
  - c) Relaciones nuevas a clases u objetos
3. Crear predicados para eliminar:
- a) Clases u objetos
  - b) Propiedades específicas de clases u objetos
  - c) Relaciones específicas de clases u objetos
4. Crear predicados para modificar:
- a) El nombre de una clase u objeto
  - b) El valor de una propiedad específica de una clase u objeto
  - c) Con quién mantiene una relación específica una clase u objeto

Para todos los predicados deberás:

- Tomar en cuenta el mecanismo de herencia con defaults y excepciones.
- Utilizar el criterio de especificidad para resolver el problema de la no-monotonicidad.
- Tomar en cuenta la interpretación de las relaciones explicada en clase.
- Cargar toda la base de conocimiento del archivo como un objeto en una sola variable (específicamente, en una lista), y utilizarla de esta manera durante toda la consulta. Al finalizar, si hubo modificaciones de la base, almacenar el resultado sobrescribiendo en el archivo del que se leyó.
- Las propiedades y las relaciones estarán en listas atributo-valor.
- Manejar información incompleta (el sistema debe responder sí, no o no sé).

Observaciones:

- Como parte de la evaluación, en la clase del 13 de octubre cada equipo tendrá que explicar brevemente su proyecto, y se le pedirá que realice algunas consultas y modificaciones para demostrar su funcionamiento.
- Trabajar en equipos de dos personas.
- Utilizar SWI Prolog versión 7.1.20 o superior