

Metodología de Programación I

Presentación del curso.

Dr. Alejandro Guerra-Hernández
Departamento de Inteligencia Artificial
Universidad Veracruzana
Facultad de Física e Inteligencia Artificial
Sebastián Camacho No 5, Xalapa, Ver., México 91000
aguerra@uv.mx
www.uv.mx/aguerra

21 de agosto de 2009

1. Objetivos

El tema principal de este primer curso de metodologías de programación orientadas a la Inteligencia Artificial es la programación lógica. El curso tiene como objetivos:

1. Presentar la lógica de primer orden como un formalismo universal de representación y procesamiento de conocimiento, ampliamente utilizado en la Inteligencia Artificial.
2. Presentar a la programación lógica como un modelo de programación de sistemas inteligentes que integran la representación y el procesamiento lógico con estrategias de búsqueda en el espacio de soluciones de problemas.
3. Explicar el desarrollo de algunas extensiones de la programación lógica, importantes para la Inteligencia Artificial, como la meta-programación, programación por restricciones y la programación orientada a agentes.

2. Evaluación

El estudiante será evaluado conforme a su **participación** en clase, la calidad de sus **tareas** y el resultado que obtenga en el **examen** final. La nota final, será calculada, aproximadamente, de la siguiente forma:

- 4 tareas/proyectos para el 60 % de la nota final.
- El examen final para el 30 % de la nota final.

- La participación en un debate y/o exposición, así como la participación durante el curso para el 10 % de la nota final.

Para obtener una nota aprobatoria, el alumno deberá haber obtenido notas aprobatorias en cada una de las tareas a realizar, el debate y el examen final, esto es, **no aprobar** cualquier elemento de la evaluación, implica no aprobar el curso.

3. Tareas

Durante el curso, se asignarán 4 tareas. Las tareas deberán entregarse a las 11:00 am del día designado (generalmente al encargarse la tarea siguiente, por ejemplo, la tarea T1 se entregará el día que será encargada la tarea T2, y así sucesivamente). El mérito del trabajo **decrece** 25 % por cada 24 horas de retraso. Las tareas pueden requerir investigación bibliográfica y experimentación en la computadora.

El formato de los **reportes** de tarea debe ser igual al que sigue este documento. La fuente en Latex del mismo se encuentra en mi página web. Si los resultados involucran código en Prolog, este debe ser **comentado** adecuadamente. La **fecha de entrega** de las tareas aparece en la sección de calendario de este documento.

La discusión entre los estudiantes es uno de los aspectos más enriquecedores en un programa de maestría. Es sumamente importante que discutan entre ustedes las tareas, pero toda discusión al respecto deberá ser de manera oral. En ningún momento podrá un estudiante consultar las notas de trabajo escrito de otro. Cualquier anomalía en este sentido, causa **expulsión** definitiva del curso.

4. Debates y Exposiciones

Cada alumno deberá participar en una exposición y/o debate, durante el curso. Tanto las exposiciones, como los debates, están relacionados con lecturas suplementarias al contenido del curso. En el caso de los debates, la postura a defender será asignada por el instructor del curso, quién fungirá como moderador. El ganador será designado por el público asistente. Los debates y o exposiciones se llevarán a cabo en la última semana del curso (martes 17 de Noviembre).

5. Examen

El examen final tendrá una duración máxima de dos horas y será de carácter escrito e individual. El examen se llevará a cabo el jueves 19 de Noviembre del presente. Pueden usar sus notas individuales para el examen, pero no las notas oficiales del curso.

6. Material del Curso

El estudiante es libre de elegir la implementación de Prolog que desee, aunque se recomienda ampliamente usar **SWI Prolog**¹ (instalado en las estaciones de trabajo ligadas a Kylie, en el centro de cómputo) o YAP².

Se les solicita usar **Latex**³ para generar los reportes con los resultados de sus tareas. Tanto Latex como SWI-Prolog pueden integrarse al editor **Emacs**⁴. El paquete AucTex que permite el trabajo con Latex, generalmente ya está instalado en las distribuciones de Emacs. El modo de trabajo para Prolog es muy limitado, por lo que se recomienda instalar un nuevo modo siguiendo las instrucciones de Bruda.CA⁵. Como ejemplo, a continuación les muestro las líneas de configuración de mi archivo `.emacs` para acceder a SWI-Prolog:

```
;;; Configuración de Emacs para Prolog

;;; Actualizar el Path con la dirección donde está el paquete elisp
(add-to-list 'load-path "/Users/aguerra/Documents/elisp/prolog/")

;;; Definir comandos para invocar a Prolog
(autoload 'run-prolog "prolog" "Start a Prolog sub-process." t)
(autoload 'prolog-mode "prolog" "Major mode for editing Prolog." t)

;;; Definir el sistema prolog usado: SWI
(setq prolog-system 'swi)

;;; Asociar extensión "pl" a los programas Prolog
(setq auto-mode-alist (append '(("\\.pl$" . prolog-mode))
                              auto-mode-alist))

;;; Incluir un índice de predicados definidos en el menú de Emacs.
(add-hook 'prolog-mode-hook 'imenu-add-menubar-index)
```

Otra opción es usar el ambiente de desarrollo **Eclipse**⁶ e instalar el *Prolog Development Tool* (PDT)⁷. Todas estas herramientas son gratuitas y pueden encontrarse en la web. Las referencias bibliográficas se listan al final de este documento, todas ellas pueden ser consultadas en mi oficina o en la biblioteca del DIA.

El material complementario, como algunos artículos, les será enviado por correo electrónico oportunamente.

7. Calendario

Las sesiones del curso se llevarán a cabo martes y jueves. Son sesiones de dos horas. El contenido de las mismas es como sigue:

¹<http://www.swi-prolog.org>

²<http://www.ncc.up.pt/vsc/Yap/index.html>

³<http://www.latex-project.org/>

⁴<http://www.gnu.org/software/emacs/>

⁵<http://bruda.ca/emacs-prolog/>

⁶<http://www.eclipse.org/>

⁷<http://sewiki.iai.uni-bonn.de/research/pdt/start>

Fecha	Tema	Tarea
25/08/2009	Introducción a la Programación Lógica	
27/08/2009	Preliminares (Lógica Primer Orden).	
01/09/2009	Programas Lógicos Definitivos	T1
03/09/2009	Resolución SLD	
08/09/2009	Resolución SLD	
10/09/2009	Negación en la Programación Lógica	
15/09/2009	Corte y Aritmética	T2
17/09/2009	Prolog	
22/09/2009	Recurrencia	
24/09/2009	Búsquedas en espacios de soluciones	
29/09/2009	Búsquedas en espacios de soluciones	T3
01/10/2009	Sistemas expertos	
06/10/2009	Sistemas expertos	
08/10/2009	Planeación	
13/10/2009	Planeación	T4
15/10/2009	Meta programación	
20/10/2009	Meta programación	
22/10/2009	Programación Lógica por Restricciones	
27/10/2009	Programación Lógica por Restricciones	
29/10/2009	AgentSpeak(L)	
03/11/2009	AgentSpeak(L)	
05/11/2009	Exposiciones	
17/11/2009	Entrega de Proyectos	
19/11/2009	Examen	

Referencias

- [1] I. Bratko. *Prolog programming for Artificial Intelligence*. Addison-Wesley, 1986.
- [2] W. F. Clocksin. *Clause and Effect: Prolog Programming for the Working Programmer*. Springer-Verlag, Berlin Heidelberg, 1997.
- [3] W. F. Clocksin and C. S. Melish. *Programming in Prolog, using the ISO standard*. Springer-Verlag, Berlin-Heidelberg, Germany, 2003.
- [4] M. Genesereth and N. Nilsson. *Logical Foundations for Artificial Intelligence*. Morgan Kaufman Publishers, Inc., Palo Alto, CA., USA, 1987.
- [5] S.-H. Nenuhuys-Chen and R. de Wolf. *Foundations of Inductive Logic Programming*, volume 1228 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, Berlin Heidelberg, 1997.
- [6] U. Nilsson and J. Maluszynski. *Logic, Programming and Prolog*. John Wiley & Sons Ltd, 2nd edition, 2000.

- [7] J. V. Leeuwen, editor. *Handbook of Theoretical Computer Science*. MIT Press, 1998.
- [8] S. J. Russell and P. Norvig. *Artificial Intelligence, a modern approach*. Prentice Hall, New Jersey, USA, 2nd edition, 2003.
- [9] L. Sterling and E. Shapiro. *The Art of Prolog*. MIT Press, Cambridge, MA., USA, 1999.