Universidad de Buenos Aires
Facultad de Ingeniería

Teoría de Lenguajes (75.31)

Trabajo Práctico

Prolog

Introducción al lenguaje y a la programación lógica

Integrantes

Axel Straminsky	XXXXX	axel_stram@hotmail.com
Demian Ferrerio	88443	epidemian@gmail.com
Martín Paulucci	88509	martin.c.paulucci@gmail.com

23 de junio de 2011

Índice

1. Paradigma Lógico

1.1. Introducción

El paradigma de la programación lógica surge de la necesidad de expresar formalmente los objetivos de un sistema, así como los conocimientos y las suposiciones acerca del mismo, a la hora de programar. La lógica provee las bases para deducir consecuencias a partir de premisas, para verificar o falsificar una proposición a partir de otras y para verificar la valiz de de un argumento lógico.

Si bien el propósito de las computadoras es ser usadas por personas, la dificultad de su construcción fue tan grande que los lenguajes utilizados para expresar los problemas que estas debían resolver fueron diseñados desde la perspectiva del funcionamiento de la computadora en sí.

Las computadoras modernas están basadas en la arquitectura de von Neumann [1], el la cual un programa consiste en una serie de instrucciones que operan sobre registros y que se ejecutan una tras otra, pudiendo la ejecución de una instrucción influir en qué ejecución se ejecute a continuación.

6. Ejemplos complejos

6.1. Cambio

6.2. N Reinas

7. Conclusiones

8. Hisotria

Prolog es un lenguaje de programación lógico ideado en los '70 por Colmerauer y Roussel.

La base conceptual del paradigma detrás del lenguaje fue desarrollada en forma independiente por Alfred Horn en los '50, al publicar *On sentences which are true of direct unions of algebras*, en la cual presenta un modelo lógico para el tratamiento de oraciones del lenguaje natural, donde se explican las luego denominadas "cláusulas de Horn".

Uno de los objetivo al crear este lenguaje fue el de poder crear programas que sirviesen como demostradores automáticos de teoremas.

Inicialmente era un lenguaje interpretado, hasta que en 1983 se desarrollo un compilador capaz de traducir Prolog a un conjunto de instrucciones de una máquina abstracta denominada Warren Abstract Machine (WAM), y desde entonces Prolog es semi-interpretado.

9. Programación Lógica

La programación lógica es un paradigma de programación dentro del paradigma de programación declarativa, y permite abstraerse del "cómo" y concentrarse en el "qué" a la hora de escribir programas. Particularmente Prolog implementa un tipo particular de lógica que es la lógica de primer orden.

Este paradigma tiene como característica principal la aplicación de las *reglas de la lógica* para inferir conclusiones a partir de datos. Conociendo la información y las condiciones del problema, la ejecución de un programa consiste en la búsqueda de un objetivo dentro de las declaraciones realizadas. Esta forma de tratamiento de la información permite pensar la existencia de *"programas inteligentes"* que puedan responder, no por tener en la base de datos todos los conocimientos, sino por poder inferirlos a través de la deducción.

Un programa lógico no tiene un algoritmo que indique los pasos que detallen la manera de llegar a un resultado, sino que está formado por expresiones que describen la solución (o más precisamente, la *declaran*). De esta manera, la clave para hacer un programa lógico es poder explicitar una declaración que describa correctamente la solución

del problema.

10. Lógica de Primer Orden

La lógica de primer orden es un sistema de lógica formal usado en distintas disciplinas, como la matemática y la computación. La lógica de primer orden se distingue de la lógica proposicional por su uso de *cuantificadores* ("para todo" o "existe").

Dentro de la lógica de primer orden, se pueden distinguir 2 partes clave: la sintaxis y la semántica. La sintaxis indica la colección de símbolos que son expresiones legales dentro de la lógica de primer orden, y la semántica determina el significado que poseen estos símbolos.

Existen 2 tipos de expresiones legales: los términos, que representan objetos, y las formulas, que representan proposiciones (expresiones que pueden ser verdaderas o falsas).

Los términos y las forumulas de la lógica de primer orden son cadenas de simbolos, que forman el alfabeto del lenguaje.

El conjunto de los términos se define inductivamente como:

- Variables: cualquier variable es un término.
- Funciones: cualquier expresión de la forma $f(t1, \cdots, tn)$ es un término.

El conjunto de las formulas se define inductivamente como:

- Simbolos de predicado: si P es un simbolo de predicado n-ario y $t1, \cdots, tn$ son terminos, entonces $P(t1, \cdots, tn)$ es una formula.
- Igualdad: si $t1$ y $t2$ son términos, entonces $t1 = t2$ es una formula.
- Negación: si P es una formula, entonces P es una formula.
- Conectivos Binarios: si P y Q son formulas, entonces $P \to Q$ es una formula.
- Cuantificadores: si P es una formula y x una variable, entonces $(paratodox)P$ y $(existealgunx)P$ son formulas.

11. Warren Abstract Machine (WAM)

La WAM es una máquina abstracta diseñada para ejecutar Prolog, consistente de una arquitectura de memoria y de un conjunto de instrucciones. Este diseño se convirtió en el standard de facto para la construcción de compiladores Prolog.

12. Aplicaciones del Paradigma Lógico

Uno de los principales campos de aplicación de este paradigma es en la Inteligencia Artificial, especialmente en todo lo relacionado con los Sistemas Expertos.

Un sistema experto es un programa que imita el comportamiento de un experto humano. Por lo tanto, contiene información (una base de conocimientos), y una herramienta para comprender las preguntas y encontrar las respuestas en la base de datos (un motor de inferencia).

También se aplica al Procesamiento Natural del Lenguaje, en donde se trata de dividir el lenguaje en partes y relaciones para tratar de comprender su significado. Otros casos en donde se utiliza este paradigma son:

- Bases de datos deductivas.
- Validación automática de programas.
- Programación Distribuida y Multiagente.
- Paralelización Automática de Programas.

13. Implementaciones del Paradigma Lógico

- SWI-Prolog: Soporta mulithreating.
- Mercury: Mezcla de mrogramación lógica y funcional.
- Fprolog: Añade lógica difusa.
- Prolog+: Añade clases y jerarquías de clases.
- LogTalk: añade *POO*.
- λprolog: soporta polimorfismo y programación de alto orden.

14. Programas lógicos

bla

15. Conclusión

Acá va la conclusión...
Así se hace una cita [2].

Referencias

[1] *Von Neumann Architecture*, artículo en Wikipedia: http://en.wikipedia.org/wiki/Von_Neumann_architecture

[2] Leon Sterling & Ehud Shapiro, *The Art of Prolog: Advanced Programming Techniques*. The MIT Press, 2nd Edition, 1999.