

PROGRAMACIÓN DECLARATIVA

PROGRAMACIÓN LÓGICA

Tema PL2: El lenguaje PROLOG, aspectos básicos

1. Características Generales

Grado en Ingeniería Informática

URJC

Ana Pradera

Contenido

- 1 EVOLUCIÓN HISTÓRICA
- 2 COMPONENTES DE PROLOG
- 3 ENTORNOS DE PROGRAMACIÓN EN PROLOG
- 4 ESQUEMA GENERAL DE TRABAJO EN SWISH PROLOG

EVOLUCIÓN HISTÓRICA

- **PROLOG** (del francés, PROgrammation en LOGique), es el primer lenguaje de programación basado en el paradigma de la Programación Lógica.
- Su **primer intérprete lo implementa en 1972** un equipo dirigido por **A. Colmerauer** de la Universidad de Marsella (Francia), utilizando resultados teóricos aportados, entre otros, por **R. Kowalski** (Universidad de Edimburgo, Escocia).
- El lenguaje tiene una rápida expansión, sobre todo en Europa y Japón, definiéndose el **estándar ISO en 1995**.
- PROLOG **no es un lenguaje declarativo “puro”**, ya que por razones de eficiencia incorpora el uso de elementos extra-lógicos con efectos laterales (para aritmética, entrada/salida, control en la búsqueda de soluciones, ...).

COMPONENTES DE PROLOG

Recuerde (**Tema PL1**) que todo entorno de Programación Lógica consta de dos componentes: un *lenguaje lógico* para la representación del conocimiento y un *sistema de Demostración Automática* para la resolución de problemas y el cómputo de soluciones.

PROLOG utiliza la conocida como **Programación Lógica Definida**:

- Su lenguaje es un subconjunto de la Lógica de Primer Orden, cuyas fórmulas se conocen como **cláusulas de Horn** (**A. Horn**, Universidad de California, EEUU) y cuya *sintaxis* se define en el siguiente **apartado**.
- Su sistema de Demostración, **Resolución SLD**, es una especialización del Sistema de Resolución de la Lógica de Predicados circunscrito a cláusulas de Horn. Su *semántica operacional* se describe en detalle **más adelante**.

ENTORNOS DE PROGRAMACIÓN EN PROLOG

- Existen múltiples entornos de programación en PROLOG, algunos comerciales y otros de libre distribución.
- La mayoría de ellos se adaptan al estándar ISO y ofrecen tanto intérprete como compilador.
- Entorno recomendado: **SWI-Prolog**
 - dominio público
 - multi-plataforma (Linux, MS-Windows, MacOS X)
 - compatible con el estándar ISO
 - incluye intérprete, compilador y múltiples bibliotecas
 - ofrece una herramienta online que no requiere instalación, **SWISH**.
Esta herramienta no incluye todo el lenguaje, pero es suficiente para la realización de todos los ejercicios y prácticas de la asignatura.

ESQUEMA GENERAL DE TRABAJO EN SWISH PROLOG

PASO 1. Escritura de un programa lógico

- Un programa PROLOG es conjunto de fórmulas lógicas adaptadas a la sintaxis específica del lenguaje (que se detalla más adelante).
- Comentarios:
 - Para comentarios en una sola línea basta con anteponer el símbolo **%**.
 - Los comentarios en varias líneas empiezan con **/*** y terminan con ***/**.
- En **SWISH** los programas se crean (botón azul “Create a program”) y editan directamente en la **parte izquierda de la ventana**.

PASO 2. Escritura de una consulta

- Para usar un programa hay que escribir **consultas**, fórmulas lógicas con una cierta sintaxis (que se detalla más adelante).
- En **SWISH** las consultas se escriben en la **zona inferior derecha** de la ventana y se activan con el botón “Run!” (esquina inferior derecha) o mediante la combinación de teclas CTRL+ENTER.
- Al recibir una consulta, lo primero que hace el sistema es *interpretar* tanto la consulta como el programa escrito en la parte izquierda.
- Posibles mensajes de error:
 - 1 Errores sintácticos: corregir errores y volver a activar la consulta.
 - 2 Avisos (warnings): repasar código en busca de posibles erratas.

PASO 3. Resolución del problema

- Si no hay errores sintácticos ni de ejecución, PROLOG pone en marcha su sistema de Demostración Automática y la respuesta, que aparece en la **parte superior derecha**, puede ser:
 - **false** si la consulta no es una consecuencia lógica del programa.
 - **true** si la consulta es una consecuencia lógica del programa pero es una mera comprobación (no hay nada que computar).
 - **los valores que hacen cierta la consulta**, si esta es una consecuencia lógica del programa y además es de tipo existencial. Si hay varias soluciones posibles, el sistema ofrece la primera y unos botones para obtener más (Next para la siguiente solución, 10/100/1000 para ver las 10/100/1000 siguientes soluciones y STOP para dejar de computar soluciones).
- Para entrar en **modo depuración** basta con anteponer la palabra `trace`, (seguida de una coma) a la consulta y utilizar los botones que ofrece el sistema para seguir la ejecución paso a paso.

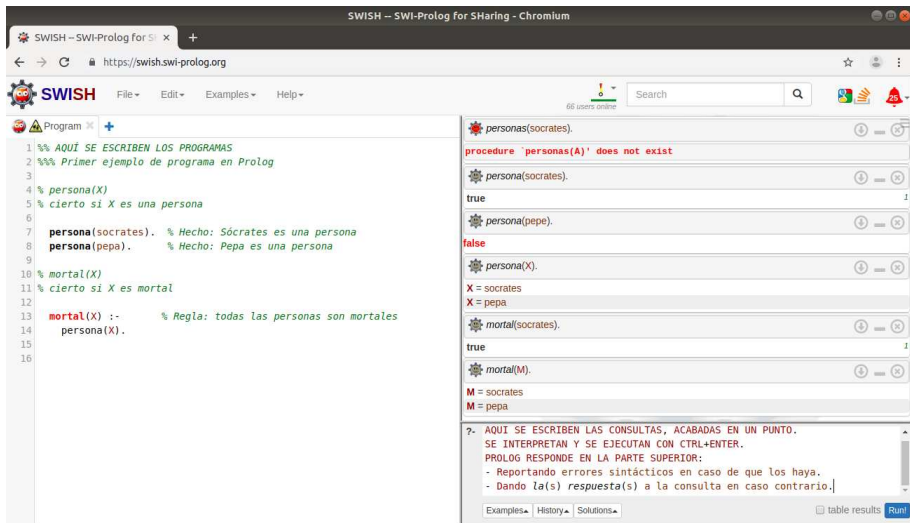


Figura: Captura de una sesión de trabajo en SWISH Prolog

Ejercicios (Primer contacto con SWISH: la mortalidad de Sócrates)

*Ponga en práctica lo explicado en este apartado escribiendo en PROLOG el problema relativo a la mortalidad de Sócrates, discutido en el tema **PL1** y recogido en la captura de pantalla anterior:*

- *Abra el entorno de programación **SWISH**.*
- *Lea la ayuda en el menú “Help”. Observe en particular que puede **registrarse** usando Google o Stackoverflow (esquina superior derecha), lo cual le dará ciertas ventajas para guardar/recuperar sus ficheros y decidir quién puede verlos/editarlos.*
- *Abra una ventana de edición de programas con el botón “Create Program” (arriba a la izquierda) y escriba el programa relativo a la mortalidad de Sócrates (ver captura de pantalla).*
- *Pruebe su programa escribiendo consultas (ver captura de pantalla) en la parte inferior derecha.*

BIBLIOGRAFÍA

- L. Sterling and E. Shapiro. *The Art of Prolog*. The MIT Press, Cambridge, Mass., second edition, 1994.
- W.F. Clocksin and C.S. Mellish. *Programming in Prolog*. Springer-Verlag, Berlin, fifth edition, 2003.
- I. Bratko. *Prolog Programming for Artificial Intelligence*. Addison-Wesley, Reading, Massachusetts, third edition, 2001.
- J. Lloyd. *Foundations of Logic Programming*, (Second Edition). Springer-Verlag, 1987.
- R. O'Keefe. *The Craft of Prolog*. The MIT Press, Cambridge, MA, 1990.
- U. Nilsson and J. Maluszynski. **Logic, Programming and Prolog**. John Wiley & Sons Ltd, 1996.
- **SWI-Prolog**, entorno de programación en Prolog de dominio público.
- **comp.lang.prolog. Faq**

© 2022 Ana Pradera Gómez

Algunos derechos reservados

Este documento se distribuye bajo la licencia

“Atribución-CompartirIgual 4.0 Internacional” de Creative Commons,
disponible en

<https://creativecommons.org/licenses/by-sa/4.0/deed.es>