



UNIVERSIDAD DE GUADALAJARA



Centro Universitario de Ciencias Exactas e Ingenierías

Departamento de Ciencias Computacionales

**Seminario de Solución de Problemas de Sistemas Basados en Conocimiento**

Tarea 1

“Paradigma de Programación Declarativo”

**Profesor:**

Valdes Lopez Julio Esteban

**Sección:**

D05

**Fecha:**

17/08/2022

**Alumno:**

Sandoval Padilla Fernando Cesar

**Código:**

215685409

**Carrera:**

Ingeniería informática

## Índice

<b>Tabla de Imágenes</b> .....	2
<b>Introducción</b> .....	2
<b>Investigación</b> .....	2
<i>Paradigmas Declarativos:</i> .....	2
Funcional .....	3
Lógico .....	4
<i>Prolog:</i> .....	5
Estructura de un programa en Prolog: .....	5
Operadores Lógicos: .....	6
<b>Bibliografía</b> .....	6

## Tabla de Imágenes

Ilustración 1 Paradigmas de Programación .....	3
--	---

## Introducción

El presente documento se encargará de analizar de forma breve la definición del paradigma de programación declarativa, así como el lenguaje de programación PROLOG y sus respectivos ejemplos.

## Investigación

### *Paradigmas Declarativos:*

Describen las características que debe tener la solución. Es decir, especifican “qué” se desea obtener, pero no requieren indicar “cómo” obtenerla.

Los paradigmas declarativos se basan en desarrollar programas especificando o “declarando” un conjunto de proposiciones, condiciones, restricciones, afirmaciones, ecuaciones o transformaciones que caracterizan al problema y describen su solución. A partir de esta información el sistema utiliza mecanismos internos de control que evalúan y relacionan adecuadamente dichas especificaciones, de manera de obtener la solución. No se necesita de la puntualización de los pasos a seguir para alcanzar una solución, ni instrucciones de control que conformen algoritmos.

## Seminario de Solución de Problemas de Sistemas Basados en Conocimiento

- Estos paradigmas permiten utilizar variables para almacenar valores intermedios, pero no para actualizar estados de información. Si bien sus variables se relacionan con posiciones de memoria, no existe el concepto asignaciones destructivas.
- Como estos paradigmas especifican la solución sin indicar cómo construirla, en principio, eliminan la necesidad de probar que el valor calculado es el valor solución. En otras palabras, se basan en “qué” es necesario especificar.

Programación imperativa vs declarativa:



*Ilustración 1 Paradigmas de Programación*

Los siguientes paradigmas forman parte de los declarativos:

- Funcional

Está basado en el modelo matemático de composición funcional. En este modelo, el resultado de un cálculo es la entrada del siguiente, y así sucesivamente hasta que una composición produce el resultado deseado. Así, un programa es un conjunto de funciones que cooperan entre ellas para el logro de un objetivo común.

Principales características:

- Transparencia referencial.

El valor de cada expresión depende únicamente de sus componentes. Las variables son utilizadas para hacer referencia a valores intermedios y parámetros de las funciones, como resultados de cálculos anteriores y entrada a subsiguientes cálculos.

- Evaluación diferida.

## Paradigma de Programación Declarativo

La evaluación de las expresiones invocadas se posterga hasta el momento en que realmente sean utilizadas.

- Recursividad.

Una función recursiva se define con al menos un término recursivo, en el que se vuelve a invocar la función que se está definiendo, y algún término no recursivo como caso base para detener la recursividad.

- Listas.

Se define como una forma de organizar conjuntos de valores, están definidas de forma recursiva y pueden ser infinitas, es decir, contener tantos elementos como se requiera sin establecer un máximo.

- Funciones de orden superior.

Las funciones que reciben a otras funciones como parámetros se llaman funciones de orden superior.

### Lenguajes:

El lenguaje más típico del paradigma funcional es el LISP y existen otros lenguajes similares más modernos como GOFER y HASKELL.

- Lógico

El paradigma lógico 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.

### Principales características:

- Lógica Proposicional

Tiene que ver con las Cláusulas de Horn. Dichas cláusulas son una forma de lógica de predicados con una sola conclusión en cada cláusula y un conjunto de premisas de cuyo valor de verdad se deduce el valor de verdad de la conclusión: una conclusión es cierta si lo son simultáneamente todas sus premisas.

- Declaraciones

Por su esencia declarativa, un programa lógico no tiene un algoritmo que indique los pasos que detallen la manera de llegar a un resultado.

Es el sistema el que proporciona la secuencia de control, por lo que el programa se transforma en un conjunto de declaraciones formales que describen la solución, que

## Seminario de Solución de Problemas de Sistemas Basados en Conocimiento

deben ser correctas por definición, por lo que la corrección del programa debería estar probada automáticamente.

- Inversibilidad

Los predicados son relaciones, que al no tener predefinido una “dirección” entre sus componentes, permiten que sus argumentos actúen indistintamente como argumentos de entrada y salida.

Esta característica se denomina reversibilidad o inversibilidad.

- Backtracking

Internamente, existe un mecanismo interno llamado backtracking que actúa como control de secuencia. Durante la ejecución de un programa va evaluando y combinando las reglas lógicas de la base de conocimiento para lograr los resultados esperados.

El backtracking no termina la búsqueda de soluciones apenas logra una respuesta válida, sino que agota todas las combinaciones lógicas posibles para ofrecer como el conjunto completo de respuestas alternativas posibles a toda consulta efectuada.

### Lenguajes:

El PROLOG es el lenguaje emblemático del paradigma. Existen diferentes versiones y variantes, pero todas basadas en la misma raíz.

### *Prolog:*

Prolog es un lenguaje de programación diseñado para representar y utilizar el conocimiento que se tiene sobre un determinado dominio. Los programas en Prolog responden preguntas sobre el tema del cual tienen conocimiento.

Un programa en Prolog consiste en una base de datos de relaciones lógicas y detalles que se cumplen para la aplicación. Dicha base de datos no tiene una estructura impuesta, ni un procedimiento o clase principal. Los datos y relaciones de un programa en Prolog se escriben en un único archivo, el cual es consultado por el programa cuando se le hace una pregunta.

- La ejecución de Prolog consiste en una búsqueda en profundidad de un árbol conteniendo todas las posibles soluciones.
- Escribir un programa en Prolog consiste en declarar el conocimiento disponible acerca de los objetos, además de sus relaciones y sus reglas.
- La estructura de un programa en Prolog es lógica y directa.

### Estructura de un programa en Prolog:

La sintaxis Prolog incluye:

## Paradigma de Programación Declarativo

- Hechos: es un hecho del lenguaje cotidiano. En lenguaje normal, una proposición puede ser "hace calor".
- Reglas: las capacidades del programa. Estas son consultadas al hacer una consulta para verificar su validez o invalidez.
- Variables: Representan objetos que el mismo Prolog determina. Una variable puede estar instanciada o no instanciada.
- Consultas: es la acción de hacerle una pregunta al programa sobre la información contenida en la base de datos. Una pregunta sencilla podría ser: ¿ 'Hace Calor'. Una respuesta podría ser: si, seguido por el prompt: ?-. Un si significa que la pregunta es consistente con la información contenida en la base de datos. Si no existe suficiente información, responde: no.

### Operadores Lógicos:

- not : negación
- \+ : no probable
- , : conjunción lógica
- ; : disjunción lógica
- :- : implicación lógica
- ->: if-then-else

## Bibliografía

- IONOS. (2020, 24 febrero). Programación declarativa: cuando el qué es más importante que el cómo [Ilustración]. Digital Guide IONOS.  
<https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/programacion-declarativa/#:~:text=de%20programaci%C3%B3n%20declarativa-,%C2%BFQu%C3%A9%20es%20la%20programaci%C3%B3n%20declarativa%3F,todos%20los%20pasos%20de%20trabajo.>
- Pérez, K. M. J., Morales, G. A. Y., & Universidad de Costa Rica, escuela de Ciencias de la Computación e Informática, San José, Costa Rica. (2007). Introducción al lenguaje de programación lógica Prolog. di-mare. Recuperado 16 de agosto de 2022, de <http://www.di-mare.com/adolfo/cursos/2007-2/pp-prolog.pdf>
- Domínguez, I. & Facultad Regional Buenos Aires Universidad Tecnológica Nacional. (2005, abril). Fundamentos teóricos de los Paradigmas de Programación. cloudfront. Recuperado 16 de agosto de 2022, de <https://d1wqtxts1xzle7.cloudfront.net/42704971/FundamTeoricosParadigProg-with-cover-page-v2.pdf?Expires=1660757628&Signature=AoaQW0jMXw0pnH~8ip5TAYQi3rnKuD9h6bF0q8s5kChh3R4VvPBCeW-fN4SLVxWD3uEYFi2lv2hcUU4fQAN0qukywUc->

## Seminario de Solución de Problemas de Sistemas Basados en Conocimiento

[6fgjl0V~jdEhfPXxoOVjLY0M5GaiXEZABYQyhygTBE09dDgf0M7OTyvogp0azlHu581r  
NtUv8g47KJ4WwXw2E~VbwN6nFWL41uz4NK63xxa-  
kikTxNpiL2YQdCVTfkkMrEIWT74qYH8tTlsgrLwvpVZ0ttsCtccPgMRLuMLT4L~xleSL  
Xz~~NeSII4NgrRXRghm0MOHcw6UCg4xK-vlGtllZ4h5sHHg1pScTZ10Dz-Rosml-  
ID7xMPIg\\_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA](#)