

Paradigmas de programación

Programming Paradigms

Jaime Andrés Mejía Osorio

Ingeniería de sistemas y computación, Universidad Tecnológica de Pereira, Pereira, Colombia

jaime.mejia@utp.edu.co

Resumen— En la programación existen distintos estilos/métodos a seguir para conseguir hacer programas destinados a funciones específicas. Por medio de estas se cumplen ciertas condiciones para que estos métodos de programación cumplan su destino siéndole más útil y fácil de llevar en práctica al programador.

Palabras clave— Programación, estilos, métodos, funciones específicas, condiciones, útil, fácil

Abstract— In programming there are different styles / methods to follow to get programs for specific functions. By means of these certain conditions are fulfilled so that these programming methods fulfill their destiny being more useful and easier to put into practice the programmer.

Key Word — Programming, styles, methods, specific functions, conditions, useful, easy

estos modelos existen por su tipo y finalidad, empezaremos con el imperativo y llegaremos hasta el paradigma para la web (Hablares de 7 tipos de paradigmas en la programación):

1. Paradigma imperativo

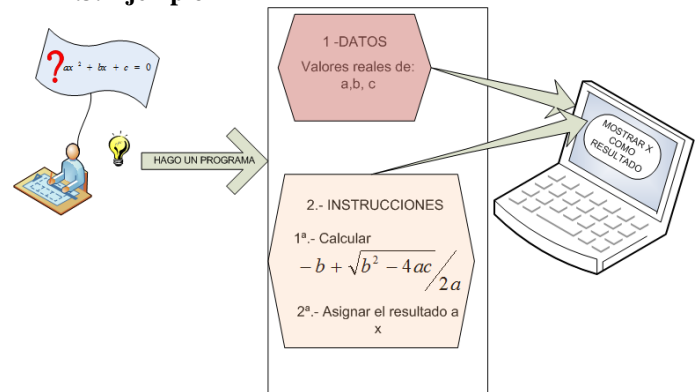
1.1. ¿Qué es?

Se puede definir fácilmente como la programación en la que se le ordena al computador y este las interpreta literalmente, siendo estas unas secuencias de comandos que cambian de estado al darle las ordenes de acción a la computadora.

1.2. ¿Cómo nació?

El hombre empezó a ver sus límites y pensó en un paradigma para expresar algunas cosas difíciles de imitar. Empezó a ampliar sus soluciones para acercarse más a la realidad.

1.3. Ejemplo



I. INTRODUCCIÓN

La palabra paradigma viniendo del griego παράδειγμα, significando en esa lengua “modelo” o “ejemplo”, nos da una idea principal de hacia donde se enfocan los paradigmas y que función tienen en el mundo de la programación (hablando de este caso, ya que el paradigma se puede adoptar en cualquier disciplina u contexto).

II. CONTENIDO

Básicamente los paradigmas de programación son la manera clara, concreta y definida de resolver un problema computacional desde unos conceptos bases; estos tienen un conjunto de reglas donde ya se da a escoger un paradigma para generar una solución en particular.

El conjunto de reglas no se cuestiona y se aceptan universalmente, siendo así un modelo de resolución de problemas a seguir de manera que no se estará improvisando de ninguna forma a la hora de generar soluciones a los problemas ya que sirven como plantilla.

En definitiva, son los modelos y métodos que se simplifican en una manera determinada para resolver un problema; y

2. Paradigma orientado a objetos

2.1. ¿Qué es?

Los objetos pasan a representarse con un tipo de dato (en informática), teniendo también atributos (características del objeto) y métodos (procesos que se asocian al mismo donde se puede programar con las características que queramos asignarle). Su funcionalidad se le hereda plenamente al objeto programado (Ya que sus características tienen su problema a resolver, con sus atributos y comportamiento único).

2.2. ¿Cómo nació?

Nació con la búsqueda de tener una aproximación a la realidad donde vivimos. Donde el mundo consta de objetos, se pensó a partir de esto que cada uno tenían sus atributos los cuales

estaban destinados a “x” función, esto se puede simular en la interfaz del computador.



3. Paradigma declarativo

3.1. ¿Qué es?

Se simplifican procesos complejos y amplios en una sola orden que el computador interpreta de manera más fácil, en su momento de creación implicaba un nivel de destreza alta para su ejecución. Es decir, si al escribir un programa estamos describiendo “Que hay que hacer”, pero no “como hacerlo”; estamos programando de forma declarativa.

3.2. ¿Cómo nació?

El pensamiento humano generó nuevas necesidades y en la búsqueda de hacer la programación algo más sencillo y efectivo, se perfilaron métodos que lo facilitarían.

3.3. Ejemplo



4. Paradigma funcional

4.1. ¿Qué es?

La concepción general consta de que un problema tiene tres principios sencillos; simplificación del objetivo (estructura de la solución a partir del cambio del panorama y es conocido como función), reutilización del código, reducción del tiempo en pruebas. El paradigma funcional está basado en conceptos que vienen de la matemática por lo que puede ser evaluado para obtener un resultado.

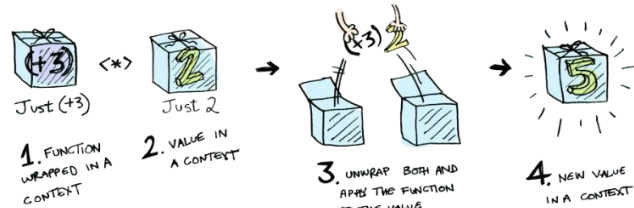
Fecha de Recepción: (Letra Times New Roman de 8 puntos)

Fecha de Aceptación: Dejar en blanco

4.2. ¿Cómo nació?

Esta forma de programación sacaba lo mejor de los otros paradigmas, pero teniendo sus características propias. Fue por Alon Church y su cálculo de Lambda que hicieron este paradigma una realidad.

4.3. Ejemplo



5. Paradigma lógico

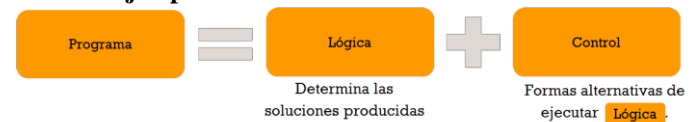
5.1. ¿Qué es?

Siendo la lógica la forma de expresar formalmente problemas complejos y de resolverlos mediante la aplicación de reglas, hipótesis y teoremas. Es básicamente darle los recursos/normas a un programa para que encuentre las soluciones a un problema.

5.2. ¿Cómo nació?

Se presenta la esencia de la lógica que ha permitido construir los paradigmas que hoy existen y su relación con la misma evolución del pensamiento.

5.3. Ejemplo



6. Paradigma por restricciones

6.1. ¿Qué es?

Tiene una relación muy clara con la programación lógica al punto que un programa en programación lógica tiene una equivalencia directa en programación con restricciones y viceversa, lo cual se hace pensando en la eficiencia. Está pensado en un conjunto de restricciones que deben ser satisfechas por cualquier solución del problema planteado, en lugar de especificar los pasos para obtener dicha solución.

6.2. Ejemplo

```
1) x es un valor entre 1 y 10
2) y es un valor entre 1 y 10
3) x < y
4) x + y = 4
```

III. CONCLUSIONES

Estas metodologías para la programación se pueden definir fácilmente como el acto del programador para solucionar un problema de manera más fácil y rápida. Estos paradigmas se diferencian de una programación sola y personal por la facilidad del uso sobre la eficiencia optima de un programa y el uso de paradigmas de programación donde el código pasa a ser más sencillo y comprensible (siendo este último un mayor avance para la capacidad cognitiva de un humano); estos códigos y programas pasan a ser no exclusivos de una sola maquina o SO, y pasan a tener la posibilidad de ser compartidos; y la complejidad del código se disminuye considerablemente.

RECOMENDACIONES

A la hora de programar está claro que habrá muchísimas formas de conseguir lo que se busca hacer y es por esto que el uso, a la vez que el conocimiento de los paradigmas (junto para que sirven), podrán resumir el código y ayudar mucho al programador para poder conseguir el objetivo de manera clara y más enfocada.

REFERENCIAS

[1]. O. I. Trejos Buritica, "Consideraciones sobre la evolución del pensamiento a partir de los paradigmas de programación de computadores" *Tecnura*, 2011.