



Instituto Tecnológico Superior de Jerez

Ingeniería en Sistemas Computacionales

Programación Lógica y Funcional

10 semestre

Tema 2.- Modelo de Programación Funcional

Mapa Conceptual.- Evaluación Perezosa

Alumno: Esteban Faustino Muñoz Hidalgo

E-mail: faustino10.96@gmail.com

No. De Control: S15070115

MTI. Salvador Acevedo Sandoval

06/03/2020.

Evaluación Perezosa

¿Qué es?

Evita la evaluación de expresiones si no son necesarias

Ventajas de su implementación:

- * El incremento en el rendimiento al evitar cálculos innecesarios, y en tratar condiciones de error al evaluar expresiones compuestas.

- * La capacidad de construir estructuras de datos potencialmente infinitas.

- * La capacidad de definir estructuras de control como abstracciones, en lugar de operaciones primitivas.

- * La evaluación perezosa puede también reducir el consumo de memoria de una aplicación, ya que los valores se crean solo cuando se necesitan.

Desventajas de su implementación:

- * Es difícil de combinar con las operaciones típicas de programación imperativa, como el manejo de excepciones o las operaciones de entrada/salida, porque el orden de las operaciones puede quedar indeterminado.

- * La evaluación perezosa puede conducir a fragmentar la memoria.

Ejemplo en Scala:

```
def calculateData: String = {  
  print("Calculating expensive data!")  
  "some expensive data"  
}  
  
def dumbMediator(preconditions: Boolean, data: String):  
  Option[String] = { print("Applying mediator") preconditions match {  
    case true => Some(data)  
    case false => None  
  }  
}  
  
def smartMediator(preconditions: Boolean, data: => String):  
  Option[String] = { print("Applying mediator") preconditions match {  
    case true => Some(data)  
    case false => None  
  }  
}  
  
smartMediator(preconditions = false, calculateData)  
dumbMediator(preconditions = false, calculateData)
```

- * smartMediator llamada a smartMediator devolverá el valor Ninguno e imprimirá el mensaje "Applying mediator".

- * dumbMediator llamada a dumbMediator devolverá el valor Ninguno e imprimirá el mensaje "Calculating expensive data! Applying mediator".

Funciones de alto orden:

Una función es de orden superior si toma una función como argumento o devuelve una función como resultado.

Usos:

- * Definición de patrones de programación.

- * Diseño de lenguajes de dominio específico.

- * Uso de las propiedades algebraicas de las funciones de orden superior para razonar sobre programas.

¿Qué es una Mónada?

Una mónada es un patrón de diseño funcional que tiene su origen en la teoría de las categorías, aunque no hace falta ser ningún experto en matemáticas para utilizarlas.

Una mónada siempre encapsula un tipo de datos que nosotros elegimos para crear instancias de otro tipo nuevo asociado a una computación especial.

¿Qué es un Monoide?

Un elemento es un monoide si cumple las siguientes premisas:

- * **Operación binaria interna:** Al combinar dos elementos del mismo tipo se obtiene otro nuevo elemento del mismo tipo de los elementos que han sido combinados.

- * **Propiedad asociativa:** No importa el orden en el que se realice la combinación.

- * **Elemento identidad:** Si al realizar la combinación de un elemento con el elemento "identidad" obtiene el mismo elemento.

(Gayo, s.f.), (itpn, s.f.), (riptutorial, s.f.), (Jiménez, s.f.), (joragupra, 2018), (Vicente, 2017)

Referencias:

Gayo, J. E. (s.f.). Programación Funcional. Obtenido de Programación Funcional:
<http://di002.edv.uniovi.es/~labra/FTP/PLF/pf.pdf>

itpn. (s.f.). itpn.mx. Obtenido de itpn.mx:
<http://itpn.mx/recursosisc/8semestre/programacionlogicayfuncional/Unidad%20III.pdf>

Jiménez, J. A. (s.f.). cs.us. Obtenido de cs.us:
<https://www.cs.us.es/~jalonso/cursos/i1m/temas/tema-7.html>

joragupra. (14 de enero de 2018). joragupra. Obtenido de joragupra:
<http://joragupra.com/2018/01/que-es-una-monada.html>

riptutorial. (s.f.). riptutorial. Obtenido de riptutorial:
<https://riptutorial.com/es/scala/example/28352/argumentos-perezosos-de-evaluacion>

Vicente, C. (10 de agosto de 2017). gigigo. Obtenido de gigigo:
<https://gigigo.com/2017/08/10/entiende-programacion-funcional-sin-ser-matematico/>