



Instituto Tecnológico Superior de Jerez

Jerez de García Salinas, Zacatecas.

Fecha: 06/03/2020

Alumno: Adán Ruiz Villalobos

Núm. Control: 16070137

Correo: adnruiz1@gmail.com

Ing. Sistemas Computacionales

Materia: Programación Lógica & Funcional.

Semestre: 8°

Actividad: Mapa Conceptual

Docente: I.S.C. Salvador Acevedo Sandoval

1. ¿Qué es la evaluación perezosa?

Estrategia de evaluación en la que una expresión no se evalúa hasta su primer uso, es decir, posponer la evaluación hasta que se solicite.

2. ¿Qué ventajas tiene la implementación de la evaluación perezosa?

- Optimiza el proceso de computación.
- La evaluación diferida puede ayudarnos a resolver dependencias circulares.
- Da acceso a una estructura de datos infinita.
- Permite modularidad de código en partes.
- El programador pierde el control sobre la secuencia en que se ejecuta su código a medida que se evalúan algunas expresiones y otras no dependen de la necesidad.

3. ¿Qué desventajas tiene la implementación de la evaluación perezosa?

- Encontrar errores puede ser complicado ya que el programador no tiene control sobre la ejecución del programa.
- Puede aumentar la complejidad del espacio ya que todas las instrucciones (operacionales) deben almacenarse.
- Más difícil de codificar en contraste con el enfoque convencional.

4. Ejemplo de aplicación de programación con evaluación perezosa en Scala

```
68 //Without Lazy Evaluation
69 var geeks = List(1, 2, 3, 4, 5)
70 val output = geeks.map(i => i * 2)
71 println(output)
72
73 //With Lazy Evaluation
74 lazy val output2 = geeks.map(i => i * 5)
75 println(output2)
76
77 } //Main
78 }
```

Console

<terminated> Prueba\$ (3) [Scala Application] C:\Program Files\Java\jre1.8.0_...
List(2, 4, 6, 8, 10)
List(5, 10, 15, 20, 25)

(GeeksForGeeks, 2017)

5. ¿Qué son las funciones de alto orden?

Una función de alto orden, puede tener una de tres formas:

- Uno o más de sus parámetros es una función y devuelve algún valor.
- Devuelve una función, pero ninguno de sus parámetros es una función.
- Ambos de los anteriores: uno o más de sus parámetros es una función, y devuelve una función.

6. ¿Qué son las MONADAS en programación funcional?

Una mónada es un patrón de diseño funcional que tiene su origen en la teoría de las categorías, Una mónada siempre encapsula un tipo de datos que nosotros elegimos para crear instancias de otro tipo nuevo asociado a una computación especial.

Sirven simple y llanamente para definir una computación un proceso que se ejecuta en un contexto.



7. ¿Qué son los MONOIDES en programación funcional?

Un monoide es cualquier tipo que tiene una operación que toma dos argumentos de ese tipo y devuelve un valor de ese tipo, asumiendo que la operación es asociativa

- $(a \text{ op } b) \text{ op } c == a \text{ op } (b \text{ op } c)$

y que también tiene un valor que no cambia el otro valor cuando se le aplica esa operación y el otro valor. Este valor se denomina de varias maneras, por ejemplo, mempty (monoide vacío) o simplemente cero, aunque un monoide no tiene por qué tener nada que ver con los números.

En otras palabras, los monoides combinan N cosas en 1 cosa, incluyendo el caso $N=0$.

Referencias

(12 de 05 de 2016). Obtenido de

https://www.reddit.com/r/scala/comments/45gqpd/whats_a_monoid/

GeeksForGeeks. (21 de 05 de 2017). Obtenido de <https://www.geeksforgeeks.org/scala-lazy-evaluation/>



