

Instituto Tecnológico Superior de Jerez  
Jerez De García Salinas, Zac.

Alumno:  
Castillo Landeros Miguel Eduardo  
Núm. Control:  
17070001

Ing. En Sistemas Computacionales.

Materia:  
Programación Lógica y Funcional.

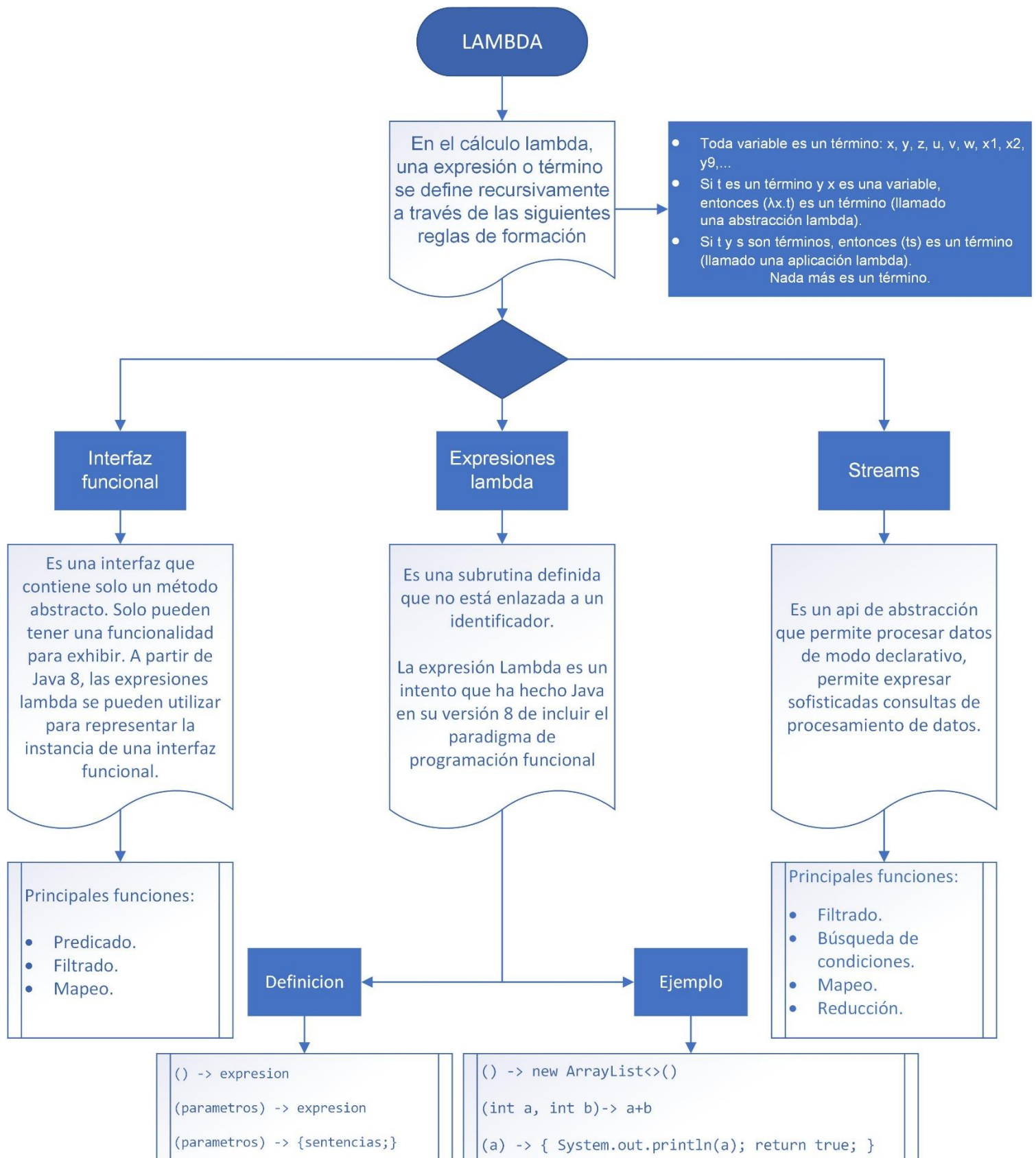
Semestre:  
8

Actividad:  
Lambda.

Docente:  
ISC Salvador Acevedo Sandoval

Fecha:  
18 de marzo 2020.

# Mapa conceptual



## 1. Matemáticamente, que es el calculo lambda

En el cálculo lambda, una expresión o término se define recursivamente a través de las siguientes reglas de formación:

- a. Toda variable es un término:  $x, y, z, u, v, w, x_1, x_2, y_9, \dots$
- b. Si  $t$  es un término y  $x$  es una variable, entonces  $(\lambda x.t)$  es un término (llamado una abstracción lambda).
- c. Si  $t$  y  $s$  son términos, entonces  $(ts)$  es un término (llamado una aplicación lambda).
- d. Nada más es un término.

## 2. ¿Qué son las 'funciones interfaces' en Java?

Una interfaz funcional es una interfaz que contiene solo un método abstracto. Solo pueden tener una funcionalidad para exhibir. A partir de Java 8, las expresiones lambda se pueden utilizar para representar la instancia de una interfaz funcional. Una interfaz funcional puede tener cualquier número de métodos predeterminados. `Runnable`, `ActionListener`, `Comparable` son algunos de los ejemplos de interfaces funcionales.

## 3. ¿Qué son las 6 interfaces funcionales del paquete `java.util.function`?

- Predicado.
- Filtrado.
- Mapeo.

## 4. ¿Qué son las expresiones lambda?

Es una subrutina definida que no está enlazada a un identificador. Las expresiones lambda a menudo son argumentos que se pasan a funciones de orden superior, o se usan para construir el resultado de una función de orden superior que necesita devolver una función.

La expresión Lambda es un intento que ha hecho Java en su versión 8 de incluir el paradigma de programación funcional, el cual es un paradigma diferente al de la programación imperativa.

Está basado en un lenguaje matemático formal.

Como ventaja tiene que es más expresivo, es decir, que nos permite usar menos código para hacer las mismas operaciones y es mucho más elegante.

## 5. Sintaxis de las expresiones lambda definicion:

Concepto:

`() -> expresion`

`(parametros) -> expresion`

`(parametros) -> {sentencias;}`

Ejemplo:

`() -> new ArrayList<>()`

`(int a, int b)-> a+b`

`(a) -> { System.out.println(a); return true; }`

## 6. Que son los streams y para qué sirve

Es un api de abstracción que permite procesar datos de modo declarativo, permite expresar sofisticadas consultas de procesamiento de datos.

Los streams permiten aprovechar las arquitecturas de núcleos múltiples sin necesidad de programar líneas de código multiproceso.

## 7. Funciones más relevantes de la clase Stream

- Filtrado.
- Búsqueda de condiciones.
- Mapeo.
- Reduccion

Referencias:

[https://www.ecured.cu/C%C3%A1lculo\\_lambda](https://www.ecured.cu/C%C3%A1lculo_lambda)

<https://www.geeksforgeeks.org/functional-interfaces-java/>

<https://docs.oracle.com/javase/8/docs/api/java/util/function/package-summary.html>

<https://openwebinars.net/blog/crear-expresiones-lambda-en-java/>

<https://www.oracle.com/technetwork/es/articles/java/procesamiento-streams-java-se-8-2763402-esa.html>

<https://docs.oracle.com/javase/8/docs/api/java/util/stream/package-summary.html>