

Programacion 2 - TUDAI - 2019

Prefinal - 28 de noviembre de 2019

Para el siguiente problema, reconocer los objetos, su estado y comportamiento, e implementar en JAVA. Tener en cuenta todos los conceptos vistos en la materia. Prestar especial atención a la **abstracción, distribución de responsabilidades, y polimorfismo.**

Punto 1

Definir un modelo orientado a objetos que permita modelar expresiones matemáticas binarias básicas: suma, resta, multiplicación, división. Por ejemplo:

- $(2+3) * 5$
- $(5 + 5) * (3 - 1)$
- $((7 + 8) / (5 - 3))$

La solución debe permitir incorporar fácilmente a futuro nuevas operaciones, como potencia y raíz.

Implementar los siguientes servicios:

- Obtener el valor de la expresión: en los ejemplos serían **25; 100; 7,5** respectivamente. Cuando el cálculo de una expresión genera un error matemático (por ejemplo, una división por 0 o la raíz de un número negativo), se debe devolver como valor resultante el número **"-23"**. Tener en cuenta que este valor puede variar a futuro, pero que debe ser siempre el mismo para todas las expresiones.
- Obtener un listado de todos los números de la expresión: en los ejemplos serían [2,3,5]; [5,5,3,1]; [7,8,5,3]
- Obtener un listado de todos los símbolos de los operadores de la expresión: en los ejemplos serían [+,*]; [+,*,-]; [+/,,-]
- Proveer una solución que permita que la siguiente línea de código:

```
System.out.println(expresión)
```

imprima por pantalla las expresiones dadas como ejemplo anteriormente tal como se muestran en el ejemplo (incluyendo los paréntesis)
- Implementar un main en el cual se crean las dos primeras expresiones dadas como ejemplo.

Punto 2

Un software de edición de expresiones matemáticas utilizará la solución propuesta en el punto 1. En este sistema, el usuario será capaz de agregar expresiones predefinidas al sistema. Se desea que el sistema pueda ofrecer un servicio de listado de todas las expresiones almacenadas, ordenadas por diferentes criterios, por ejemplo:

- Resultado del cálculo de la expresión
- Cantidad de operadores de la expresión
- Comparación del string que representa a la expresión

La solución propuesta debe ser flexible a la posibilidad de agregar a futuro nuevas formas de ordenar las expresiones, por ejemplo por cantidad de números de la expresión

Punto 3

Identificar los patrones utilizados en la solución de los puntos 1 y 2 y el lugar del diseño donde se aplicaron.