



Trabajo Final POO 2019

La empresa Wolfram Inc está pensando en desarrollar su nuevo software graficador de funciones y le ha encomendado el trabajo de diseño y programación del mismo.

El sistema debe tener la capacidad de leer una cadena de texto que representa una ecuación del modo $y = f(x)$ y a partir de ahí poder hacer diferentes operaciones sobre la función.

La cadena de texto solo representará la porción de $f(x)$ de la ecuación y debe ser interpretada y descompuesta en alguna estructura de datos que represente a dicha ecuación. Una ecuación puede tener **términos** de un solo parámetro o de dos parámetros. También debe soportar literales (números) que pueden o no tener punto decimal (es decir que es válido tanto 3 como 3.0 o 3.1). La letra 'x' tendrá un significado especial y será la variable a reemplazar en la evaluación de la función.

Se deben soportar los siguientes operadores de dos parámetros: +, -, *, /, ^ (exponente). Además se debe poder soportar los siguientes operadores de un argumento: cos(x), sin(x), tan(x), x!, sqrt(x). Los términos son combinables y pueden ser agrupados por paréntesis para generar expresiones más complejas como por ejemplo : $f(x) = 3 * \cos(x + (2 * \sin(x) - 0.25 * \cos(x^2)))$.

El trabajo deberá estar separado en diferentes partes :

- Un parser que tome la cadena de texto y convierta la cadena en alguna estructura que permita ser evaluada para un valor de x determinado
- Un módulo que permita evaluar $f(x)$ para uno o más valores de x. Este módulo deberá soportar las siguientes operaciones:
 - Evaluar $f(x)$ para un valor solo de X
 - Evaluar $f(x)$ para un intervalo [a, b] con N puntos
- Un módulo visual que grafique $f(x)$ para un intervalo [a, b] con N puntos.

Implementación

El objetivo es a partir de la expresión de la función generar una estructura que contenga **nodos**, los cuales representan los distintos tipos de operaciones que se evaluarán. Se recomienda armar una jerarquía de clases con nodo término como clase padre y a partir de ahí generar una clase para cada operación.

Para la parte gráfica se deberá utilizar la estructura de clases Morph de Smalltalk. Se necesitará mínimamente una ventana con los puntos de la función en cuestión, una caja de texto para ingresarla y los botones necesarios para interactuar con el usuario.

Referencias útiles

<https://www.codeproject.com/articles/88435/simple-guide-to-mathematical-expression-parsing>
https://en.wikipedia.org/wiki/Shunting-yard_algorithm

Condiciones de entrega

- Se deberá realizar un informe detallando la problemática con el diagrama de clases utilizado y el diagrama de secuencia del mensaje `evaluarEn: unValor` de la clase padre que represente a las operaciones y sus derivados
- La entrega deberá ser un solo archivo comprimido con informe y diagramas adjuntos en formato PDF y el código fuente exportado de Pharo (.ts)
- El nombre del archivo deberá contener los apellidos de todos los integrantes.
- Los diagramas deberán ser realizados con algún software adecuado (www.draw.io, Dia, etc).
- Finalmente se deberá defender frente a la cátedra.
- El trabajo puede ser realizado por un máximo de 3 estudiantes.