**Reporte**

**Integrantes**

1. María de Lourdes Choy Fernández

2. Javier Rodríguez Sánchez

3. Héctor Miguel Rodríguez Sosa

4. Sebastián Suárez Gómez

**Introducción**

A continuación se detallará la ficha técnica de este proyecto. El trabajo consiste en un derivador de funciones de varias variables, por una tabla de derivadas. En esencia todas las expresiones son clases implementadas usando de base la clase Expresion. Esta obliga por contrato a todas sus herederas a tener métodos para derivar, simplificarse y evaluarse; además de que internamente utiliza un sistema de prioridades asignados a cada operación o función para poder asignar mejor su declaración implícita a texto. Más adelante se detallará mejor pero por ahora se puede decir que cada función o expresión matemática se dividió para su especialización en Expresiones Unarias, Binarias, Constantes y Variables. La manera de parsear las expresiones de expresión -> string se hace directamente dentro de cada semiespecialización de la expresión.

De la manera que funciona el derivador es por variables. Si se quiere derivar una variable solo se derivará las expresiones que tengan esa variable y de una manera que no afecte al resto de variables.

El diagrama de cómo se distribuyen las expresiones ya implementadas fluye de la siguiente manera:

- Expresion

- Constant

- Variable

- Binary Expresion

- Sum

- Diference

- Multiplication

- Divition

- Potence

- Logaritm

- Unary Expresion

- Sin && asin

- Cosin && acos

- Tangent && atan

- Cotangent && acot

- Secant && asec

- Cosecant && acsc

No se va a hacer mucho hincapié en la forma de evaluarse y derivarse de cada expresión en sí ya que como se dijo anteriormente es una derivación por tabla y la evaluación se puede inducir

**Constantes y variables**

Son la forma más básica de expresión. Las constantes representan un número, mientras que las variables representan un carácter de una cadena.

Las constantes son las más sencillas, ya que su derivada es 0, su valor es el mismo número, su manera de evaluarse y de simplificarse igual. Las varibale mientras funcionan de una manera diferente, no tienen valor, su manera de simplificarse son ellas mismas, al igual que la variable a la que responden y solo se pueden evaluar si se le da un valor a ellas; y su derivada es 1 si la variable que se quiere derivar es ella y 0 si no es. Sus prioridades son ambas de 0.

Siempre que cualquier expresión se vaya a querer simplificar a si misma, va a tener que simplificar a todas las expresiones que la componen. Esto se hace en esencia evaluando y ver si se pueden anular algunos valores. Si existe una variable componiendo a esa expresión entonces la manera de simplificarse será trabajar con ellas mismas porque no puede reducirse a un valor.

**Binary Expresion**

Son las operaciones más básicas de la matemática, suma, resta, etc. Se diferencian del resto de expresiones ya que como su propio nombre lo indica son operaciones binarias, necesitan dos expresiones, una a la derecha y una a la izquierda para que puedan ejecutarse correctamente.

Estas funcionan con operadores que pueden ser suma (+), resta (-), logaritmo (log[a]), etc.

La manera de parsearse este tipo de expresiones funciona de la siguiente manera:

1. Si la prioridad de la expresión derecha es menor que la prioridad de la misma expresión y asumiendo que no sea 0, y si sucede lo mismo con la derecha entonces se interpreta como (Izquierda) + Operador + (Derecha)

2. Si sucede lo anterior pero no lo de la derecha entonces se interpreta como (Izquierda) + Operador + Derecha

3. Si no sucede nada de lo anterior entonces se interpreta como Izquierda + Operador + Derecha

4. Si sucede solo lo de la derecha entonces es Izquierda + Operador + (Derecha)

Hablemos primero de las básicas. La suma y la resta se evalúan en dependencia de su operador, o sea el resultado de su evaluación es la expresión de la izquierda +- la de la derecha. Sus derivadas son la derivada de la expresión de la izquierda +- la de la derecha. Se simplifican simplemente evaluando las expresiones de la izquierda y derecha y ver si se pueden reducir con términos, ya sean constantes o variables entre sí. Además la prioridad que reciben por defecto es de 1, que es la más baja después de las constantes

La multiplicación y la división funcionan de igual manera, salvo por su prioridad y derivada. Su prioridad es de 2. La derivada de la multiplicación es la derivada de la izquierda por la derecha más la derivada de la derecho por la izquierda.

La exponencial es donde se empiezan a ver los cambios. La prioridad varía, si su base es e entonces su prioridad es 2, pero sino, es 3. A la hora de derivar empieza a actuar la regla de la cadena. Si la base es e entonces su resultado va a ser la derivada de la expresión de la derecha por la expresión completa. Si tiene un número como exponente entonces su derivada es la derivada de la de la izquierda por el exponente por una nueva potencia hecha por la izquierda elevada a la derecha -1. No creo que haga falta en hacer énfasis en como se obtiene el valor de las expresiones de aquí en adelante.

El logaritmo irónicamente funciona de igual manera, ya que usa el número de Euler para resolver las expresiones mejor. Su prioridad es exactamente igual que la exponencial. Su derivada si es de base e, entonces es la derivada de la expresión de la derecha dividido por la expresión en sí. Sino deriva como deriva un logaritmo normal (derivada de la derecha entre la derecha por el logaritmo al revés). Si por alguna razón no entro en ninguno de estos casos entonces es derivar la división del logaritmo natural de la derecha por el logaritmo natural de la izquierda.

**Unary Expresion**

Las expresiones unarias funcionan mas o menos de la misma forma; salvo por el hecho de que en vez de tener izquierda y derecha, tienen una expresión interna y su operador es la función a la que se refieren. La manera de su conversión explícita es siempre la de su operador + (Interna).

Seno y coseno son las más básicas. Sus prioridades son de 2. Se derivan como seno -> coseno y coseno -> -seno. Aquí también se aplica la regla de la cadena. Lo mismo sucede con la tangente pero es tan -> sec^2 y la secante como sec -> sec\*tan. La cotangente deriva como -cotagente cuadrada y la cosecante como - cosecante por cotangente.

Sus inversas funcionan de la misma manera, con regla de la cadena incluida pero como sus fórmulas son tan pesadas de escribir a mano se le recomienda al lector a ver la misma tabla de derivación directa

El hecho de que la derivada sea por tabla hace el trabajo del derivador mucho más fácil. Ya que solo tiene que aprenderse ciertas reglas básicas y el resto es es utilizar la regla de la cadena, que como es una tarea fácil se puede automatizar de la misma forma.