

MINUTAS

LOGICA:

- Nuestro método `esInfija()` no reconoce todos los casos en que una expresión no es infija o no está bien parentizada.
- Tuvimos problemas para crear el `ArbolBinario` con la expresión, nuestro método `crearArbol(String s)` recibe una expresión, la transforma en un `Char[]` y luego mediante un método privado lo transforma en un `String[]` para recorrer la expresión con mayor facilidad.
- El método `resolver()` que resuelve la expresión utilizando una Pila está implementado en manera muy similar al método `crearArbol()` resolviendo siempre la subexpresión de los tres elementos en el tope de la pila.
- Creamos el método `subexpresion(Position<String> n,String s)` para ser llamado por la GUI recibiendo las posiciones ya guardadas por el método `simplificar()` y un `String` enviado por consola por el usuario para la nueva clave y nombre de las subexpresiones.

GUI:

- Tuvimos complicaciones para imprimir el listado de variables, primero tratamos de utilizar un `JList()` pero nos resulto mas sencillo utilizar un `JTextArea()`.
- Asumiré que la Expresión ingresada por consola es Infija y que las Variables creadas no tiene nombres con símbolos operando (Ej: "a+cd", "rt*e").
- Imprimiré carteles si ingresa algo no valido o trata de resolver algo que no se puede, no aseguramos que siempre que imprima un cartel haya verdaderamente un error.
- Si se intenta realizar una división por cero se escribirá la expresión en las diferentes formas, pero no se resolverá el resultado ni se creara el árbol.
- Si se desea abreviar una sub expresión entre dos variables, el programa permite nombrar es nueva con el nombre de alguna de la dos anteriores. Ej: La subexpresión "A+B" puede renombrarse y remplazarse por la variable "A" o "B".

TDAARBOLBINARIO

- Tuvimos complicaciones con la implementación del método `Attach(BinaryTree<E> T1, BinaryTree<E> T2, Position<E> p)` ya antes solo copiábamos la referencia a los arboles pasados por parámetro.
- Fue testeado y no se han encontrado errores, excepto el método antes mencionado el cual no funciona con el tester.

TDAMAP

- Tuvimos complicaciones realizando el método privado `trasladar()` para aumentar el tamaño del arreglo.
- Fue testeado y no se han encontrado errores.

TDAPILA

- Fue testeado y no se han encontrado errores.

TDALISTA

- Fue testeado y no se han encontrado errores.