

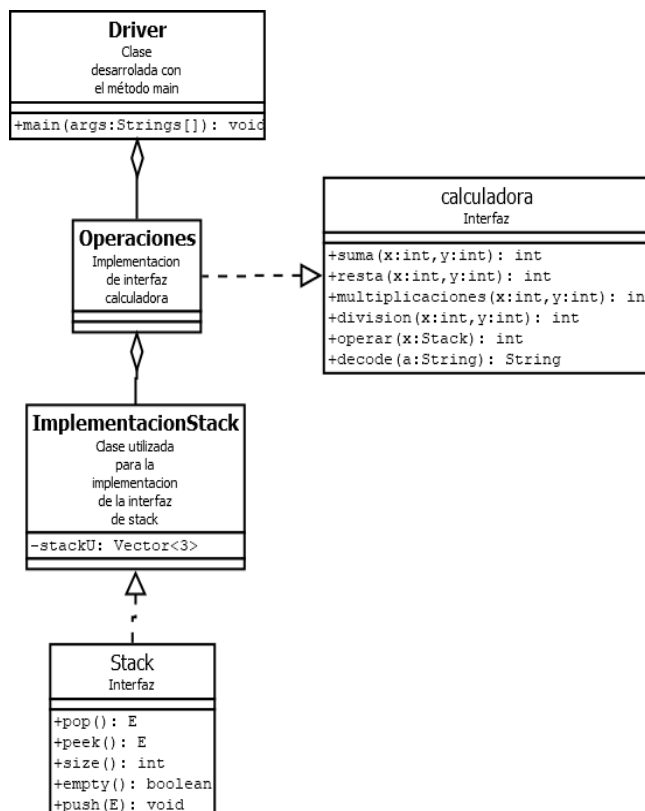
Hoja de trabajo no.2

Se realizó la hoja de trabajo 2 con el objetivo de implementar la utilización de genéricos, el diseño ADT para pilas, la utilización de pilas con un vector de tamaño variable. Así mismo, se tomó en cuenta la importancia del control de versiones del programa y la empleación de Junit.

El programa por realizar es capaz de leer un archivo de texto, con expresiones que sean del formato Postfix. Es llamado “datos.txt”, está incluido en el ZIP de esta hoja de trabajo.

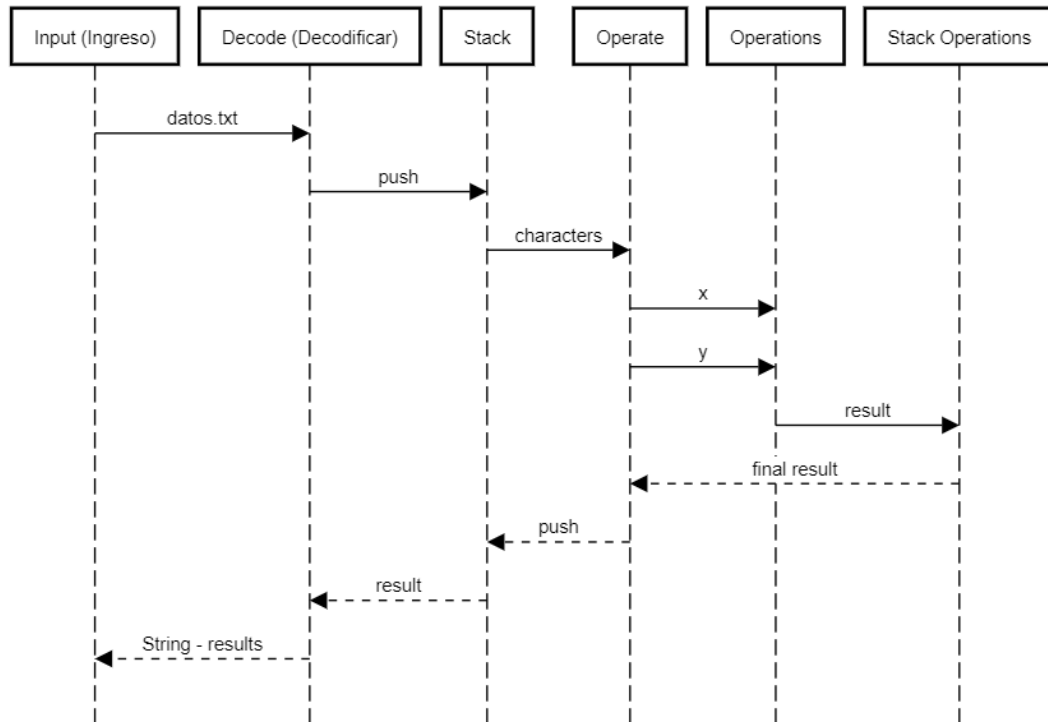
Inciso a)

UML de las clases realizadas



UML de secuencia

HT2 - Calculadora



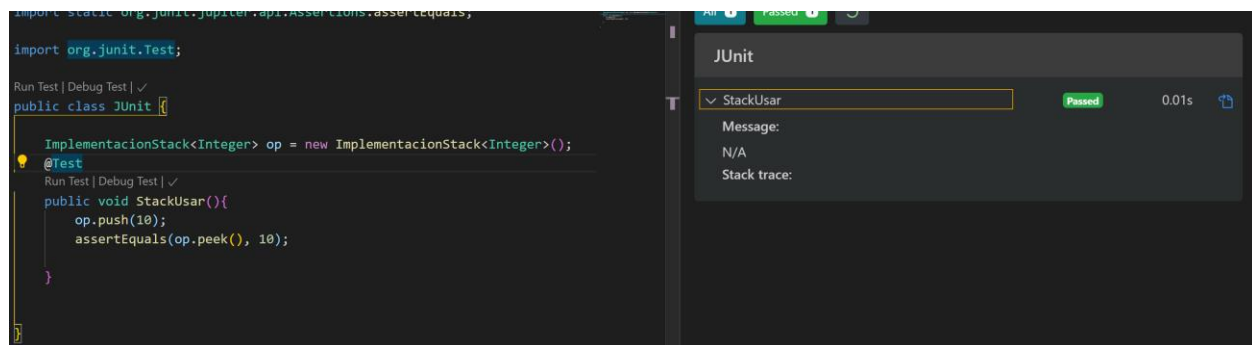
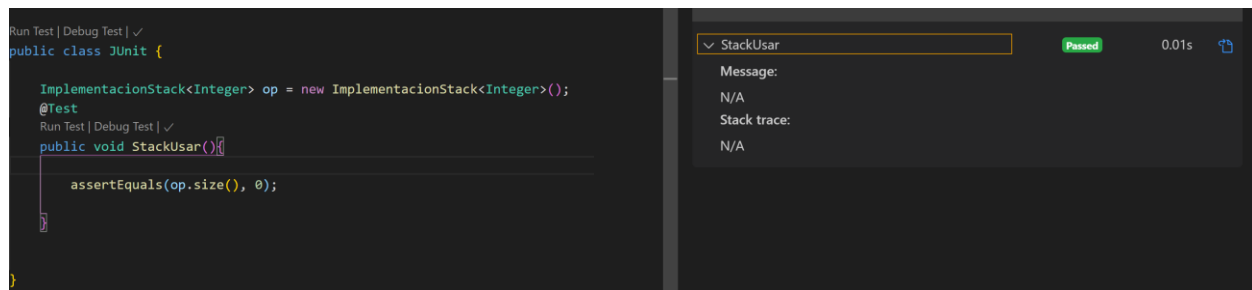
Evidencia del correcto funcionamiento del programa:

```
Operaciones:
PROCESANDO INFORMACION.....
Check check--
El dato 5 se agrego a la pila
PROCESANDO INFORMACION.....
Check check--
El dato 9 se agrego a la pila
OPERACION: Suma
Efectuado: 9 + 5
Resultado: = 14
PROCESANDO INFORMACION.....
Check check--
El dato 7 se agrego a la pila
OPERACION: Multiplicacion
Efectuado: 7 * 14
Resultado = 98
PROCESANDO INFORMACION.....
Check check--
El dato 6 se agrego a la pila
OPERACION: Suma
Efectuado: 6 + 98
Resultado: = 104
Operacion# 1: 15
Operacion# 2: 104
```

Pruebas JUnit realizadas:

The image displays five screenshots of JUnit test results in an IDE. Each screenshot shows a code editor on the left and a test results panel on the right.

- Suma:** The code defines a `JUnit` class with a `Suma()` method that calls `op.suma(5,5)` and asserts it equals 10. The test result shows "Suma" passed in 0.01s.
- Resta:** The code defines a `JUnit` class with a `Resta()` method that calls `op.resta(5,3)` and asserts it equals 2. The test result shows "Resta" passed in 0.01s.
- Division:** The code defines a `JUnit` class with a `Division()` method that calls `op.division(10,2)` and asserts it equals 5. The test result shows "Division" passed in 0.01s.
- Multi:** The code defines a `JUnit` class with a `Multi()` method that calls `op.multiplicacion(10,2)` and asserts it equals 20. The test result shows "Multi" passed in 0.01s.
- StackUsar:** The code defines a `JUnit` class with a `StackUsar()` method that calls `op.empty()` and asserts it returns true. The test result shows "StackUsar" passed in 0.01s.



Evidencia de la utilización de otras calculadoras en mi programa:

```

D:\UVG\Segundo Año\Primer semestre\Estructura Datos\HDT 2 Pruebas\HDT 2 - Prueba otra calculadora>java Driver.java
Simulador de calculadora:
Bienvenido
Escriba el nombre de su archivo
Pro tip: Solo puede ser datos.txt :)
datos
SI LLEGUE AQUI
Se ha a?adido correctamente 1 al stack
Se ha a?adido correctamente 2 al stack
La suma de los numeros 2 y 1 se ha realizado.
El resultado 3 se ha guardado en el stack.
Se ha a?adido correctamente 4 al stack
La multiplicacion de los numeros 4 y 3 se ha realizado.
El resultado 12 se ha guardado en el stack.
Se ha a?adido correctamente 3 al stack
La suma de los numeros 3 y 12 se ha realizado.
El resultado 15 se ha guardado en el stack.
-----
-----> El resultado de la operacion no. 2 es 15.
-----

```

Prueba realizada con calculadora desarrollada por Marco Jurado

Uso del repositorio:

<https://github.com/aleg001/HDT2>