**LABORATORIO 2**

**FEDERICO BARRIOS MENESES – CRISTIAN RUIZ**

**CONOCIENDO EL PROYECTO.**

1. **El proyecto BlueJ “calmatfra” contiene una construcción parcial del sistema. Revisen el directorio donde se encuentra el proyecto. Describan el contenido considerando los directorios y las extensiones de los archivos.**

* Observando el directorio donde se encuentra el proyecto, podemos ver que hay cuatro archivos con la extension ***.class*** los cuales corresponden a las clases de las que se compone el proyecto en un inicio. Luego vemos también, cuatro archivos ***.java*** los cuales contienen el código correspondiente a cada clase. Además, encontramos cuatro archivos con la extension ***.ctxt*** que contienen los parámetros que corresponden a cada una de las cuatro clases del proyecto. Por último está la carpeta ***doc*** que contiene toda la documentación del proyecto.

1. **Exploren el proyecto en BlueJ ¿Cuántas clases tiene? ¿Cuál es la relación entre ellas? ¿Cuál es la clase principal? ¿Cómo la reconocen? ¿Cuáles son las clases “diferentes”? ¿Cuál es su propósito?**

**Tiene 4 clases que son:**

* Calmatfra
* Matriz
* Fraccionario
* FraccionarioTest

La relación entre ellas es que una matriz se compone de uno o muchos fraccionarios que se representan como una tupla donde el primer elemento es el numerador y el segundo elemento es el denominador. Adicional, las pruebas de unidad se realizan sobre la clase Fraccionario y la clase Calmatfra se compone de muchas matrices y las almacena en un mapa.

La clase principal es Calmatfra y se reconoce porque es la que reparte responsabilidades a las demás clases en el diagrama de secuencia.

La clase diferente es la clase FraccionarioTest y su popósito es ejecutar las pruebas sobre la clase Fraccionario.

Para las siguientes dos preguntas sólo consideren las clases “normales”:

1. **Generen y revisen la documentación del proyecto; ¿está completa la documentación de cada clase? (Detalle el estado de documentación de cada clase: encabezado y métodos)**

* La documentación de la clase Calmatfra está incompleta, tiene encabezado pero los métodos no están documentados.
* La documentación de la clase Matriz también está incompleta, no tiene encabezado y solo algunos metodos están documentados.
* La documentación de la clase Fraccionario sí está completa.

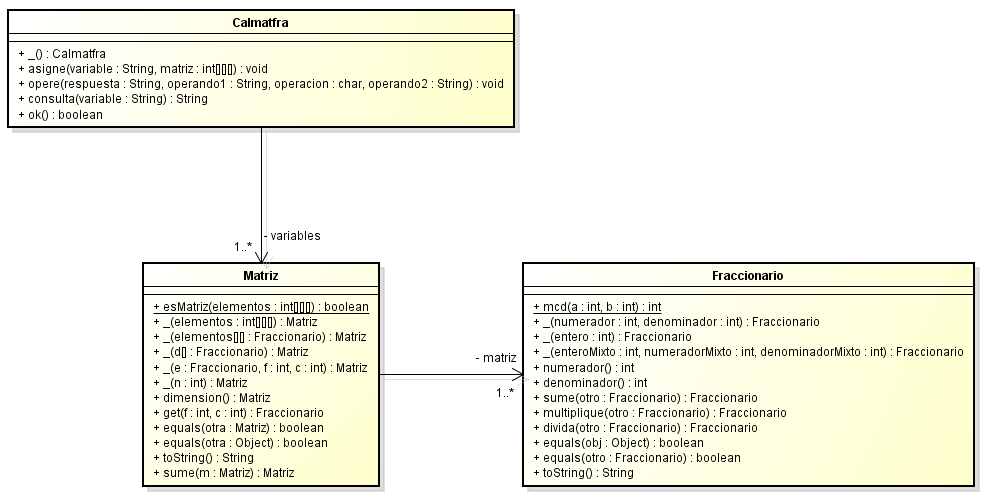
1. **Revisen el código del proyecto, ¿en qué estado está cada clase? (Detalle el estado de codificación)**

* Todas las clases están incompletas puesto que ninguna tiene todos sus métodos implementados.

**INGENIERÍA REVERSA.**

1. **Genere el diagrama de clases correspondiente a calmatfra con todos sus elementos. (No incluya la clase de pruebas)**

* El diagrama de clases correspondiente se encuentra adjunto en el laboratorio.



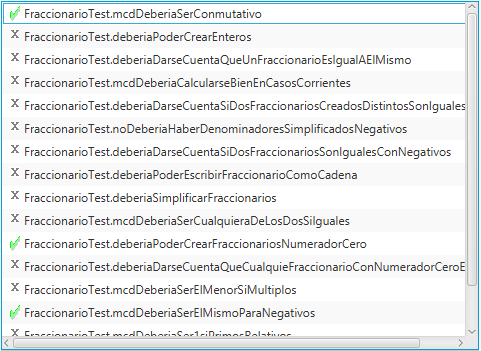
1. **¿Qué tipos de contenedores tienen sus colecciones? Consulte la especificación y el API Java 1¿Qué diferencias hay entre ellos?**

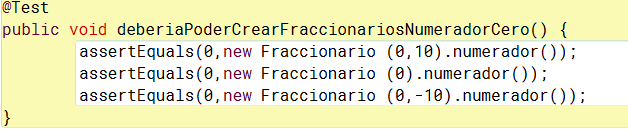
**CONOCIENDO PRUEBAS EN BLUEJ.**

1. **Revisen el código de la clase FraccionarioTest. ¿cuáles etiquetas tiene (componentes con símbolo @)? ¿cuántos métodos tiene? ¿cuántos métodos son de prueba? ¿cómo los reconocen?**

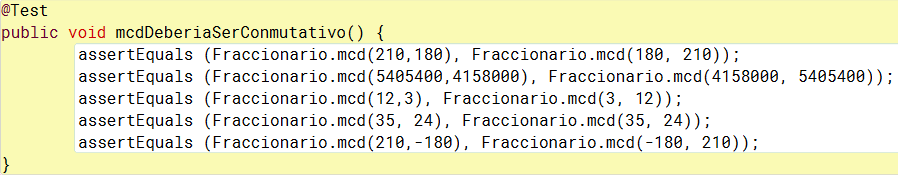
* Solamente la etiqueta @Test, tiene 16 métodos los cuales todos son de prueba. Se reconocen por la etiqueta que poseen.

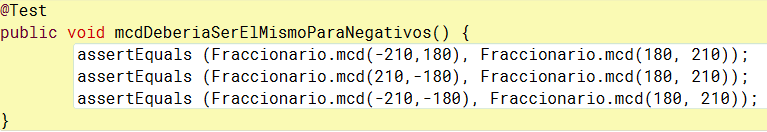
1. **Ejecuten los tests de la clase FraccionarioTest. (click derecho sobre la clase, Test All) ¿cuántos tests se ejecutan? ¿cuántos pasan las pruebas? ¿por qué?**

* Se ejecutan 16 pruebas de las cuales 3 pasan las pruebas:
* La prueba deberiaPoderCrearFraccionariosNumeradorCero() funciona debido a que el método numerador() de la clase Fraccionario retorna siempre el valor 0, con lo cual assertEquals(0,0) es una prueba exitosa.



* mcdDeberiaSerConmutativo()y mcdDeberiaSerElMismoParaNegativos() funcionan ya que sin importar sus parámetros el método mcd() de la clase Fraccionario siempre retorna 0, por lo cual siempre funciona este test.





1. **Estudie las etiquetas encontradas en 1. Expliquen en sus palabras su significado.**

* Esta etiqueta sirve para diferenciar a un método normal de un método de prueba

1. **Estudie los métodos assertTrue, assertFalse, assertEquals, assertNull y fail de la clase assert del API JUnit Explique en sus palabras que hace cada uno de ellos.**

* assertTrue sirve para ver si una condicion es true, de lo contrario envía un assertionError
* assertFalse sirve para ver si una condición es false, de lo contrario dira que hubo un assertionError enviando el mensaje que se desee
* assertEquals sirve para ver si dos objetos son iguales (si los dos objetos son nulos son considerados iguales), si no lo son se envía un assertionError con el mensaje dado
* assertNull sirve para ver si un objeto es nulo, si no lo es envía un assertionError
* fail sirve para hacer fallar un test

1. **Investiguen la diferencia que entre un fallo y un error en Junit. Escriba código usando los métodos anteriores para lograr que los siguientes tres casos de prueba se comporten como lo prometen deberiaPasar, deberiaFallar, deberiaErrar.**

* Las fallas notifican un resultado de prueba no válido, los errores indican una ejecución de prueba inesperada.

**PRACTICANDO PRUEBAS EN BLUEJ.**

**RETROSPECTIVA**

1. ¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes? (Horas/Hombre)

El tiempo invertido por cada uno de nosotros fue de aproximadamente 15 y 10 horas

1. ¿Cuál es el estado actual del laboratorio? ¿Por qué?

Podría decirse que el estado actual del laboratorio es parcialmente completado ya que falta únicamente algunas pruebas y diagramas

1. Considerando las prácticas XP del laboratorio. ¿cuál fue la más útil? ¿por qué?

Para nosotros la más útil fue la llamada “All code must have unit test” ya que nos dio buenas ideas para aplicar a nuestros trabajos en grupo como lo es la manera correcta de realizar pruebas unitarias para el código que se va a desarrollar

1. ¿Cuál consideran fue el mayor logro? ¿Por qué?

Nuestro mayor logro fue el avance que se logró en el laboratorio ya que a pesar de que tuvimos trabas o dudas durante el proceso logramos superarlas y avanzar en el desarrollo del laboratorio

1. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

El mayor problema técnico para nosotros fue que en distintas ocasiones no lográbamos entender cuál era el objetivo de un método por lo que nos consumía tiempo el intentar comprenderlo

1. ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?

Como equipo logramos comunicarnos y escuchar las ideas del otro. Nos comprometemos a mejorar el trabajo en equipo y el desarrollo de los trabajos que vienen