**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.**

**2. Exploren el proyecto en BlueJ**

**¿Cuántas clases tiene?**

Existe 3 clases en el proyecto BlueJ “calmatfra”.

**¿Cuál es la relación entre ellas?**

**¿Cuál es la clase principal?**

La clase principal es calmatfra

**¿Cómo la reconocen?**

Es aquella que no recibe una relación con dirección a ella, además los comportamientos.

**¿Cuáles son las clases “diferentes”?**

La clase FraccionarioTest es la clase “Diferente” respecto a la clase calmatfra, Matriz, Fraccionario

**¿Cuál es su propósito?**

construir pruebas para la calculadora de matrices de fraccionarios con memoria.

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

**3. Generen y revisen la documentación del proyecto;**

**¿está completa la documentación de cada clase?**

Clasecalmatfra:

La documentación no esta completa en

**Clase calmatfra:**

En esta solo se encuentra la documentación sobre la clase, en algunos de sus métodos existe una documentación sobre qué hará cada uno, pero no cumplen la documentación establecida, por ejemplo, no se menciona que significa cada uno de los parámetros recibe y que se espera que retorne.

**Clase Matriz:**

En la documentación sobre la clase se encuentra incompleta ya que solo se nos menciona el autor, en algunos de sus métodos existe una documentación sobre qué hará cada uno, pero al igual de calmatfra no cumplen algunos requisitos

**Clase Fraccionario:**

La documentación en general cumple los estándares establecidos, el único problema que se presenta es que en código se especifica que algunos métodos retornan un valor; pero en la documentación no aparece **@return**  especificando que se espera que retorne.

**(Detalle el estado de documentación de cada clase: encabezado y métodos)**

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

**Clase calmatfra:**

En esta clase solo existe el encabezado de cada uno de los métodos y además solo existe una variable de tipo HashMap.

**Clase Matriz:**

En esta clase solo existe el encabezado de cada uno de los métodos y además solo existe una variable de tipo matriz con fraccionarios llamada matriz, cabe recalcar que en algunos métodos los parámetros no son suficientes para cumplir lo que se solicita.

**Clase Fraccionario:**

En esta clase solo existe el encabezado de cada uno de los métodos y NO existe variables asociadas a la clase.

**Ingeniería reversa**

**MDD MODEL DRIVEN DEVELOPMENT**

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

Se encuentra en el LAb02.Astah

1. **¿Qué tipos de contenedores tienen sus colecciones?**

En la clase Calmatfra se tiene el contenedor **variables ( HashMap )**

En la clase Matriz se tiene el contenedor **matriz ( Array )**

**Consulte la especificación y el API Java**

**HashMap**

**“tomado de API Java 8”**

Implementación basada en un hash table. Esta implementación provee todas las operaciones opcionales del mapa, y permite valores nulos y la clave nula.

**“tomado de API Java 8”**

**Array**

Un **array** es una estructura de datos que nos permite almacenar una gran cantidad de datos de un mismo tipo

**1¿Qué diferencias hay entre ellos?**

Que hashMap tiene asociado una clave (identificador) que se le asocia a una variable y es guardada de manera **NO** sincronizada mientras que una Array almacena las variables en memoria un tras de otra.

**Conociendo Pruebas en BlueJ**

**Conociendo Pruebas en BlueJ**

**1. Revisen el código de la clase FraccionarioTest.**

**¿cuáles etiquetas tiene (componentes con símbolo @)?**

Existen 16 etiquetas (@Test)

**¿cuántos métodos tiene?**

Existen 16 métodos

**¿cuántos métodos son de prueba?**

**16 métodos so de prueba**

**¿cómo los reconocen?**

Ya que tienen etiquetas y usan el método asserEquals para compararar dos tipos, retornando un booleano que representa si son iguales o no, entre una constante y una operación realizada.

**Ejecuten los test de la clase FraccionarioTest. (click derecho sobre la clase, Test All)**

**¿cuántos test se ejecutan?**

Se ejecutan 16 test

**¿cuántos pasan las pruebas?**

Únicamente pasan 3 test

**¿por qué?**

Porque únicamente esos 3, dan el valor que se compara con el método assertEquals.

1. **Estudie las etiquetas encontradas en 1.**

**Expliquen en sus palabras su significado.**

Ya que con la etiqueta @Test se le dice al programa que se va a ejecutar una prueba, la cual se encuentra debajo de esta.

1. **Estudie los métodos**

**De la clase assert del API JUnit 2**

**. Explique en sus palabras que hace cada uno de ellos.**

**assertTrue:**

ingresa una condición y retorna si esa condición es verdadera.

**asserTrue(**boolean condition);

**assertFalse:**

ingresa una condición y retorna si esa condición es falsa.

**assertFalse**(boolean condition);

**assertEquals,**

afirma si dos objetos son iguales o no.

**assertEquals**(object1, object2);

**assertNull**

afirma si un objeto es nulo

**asserNull**(object1);

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**

**Fallo:**

El fallo ocurre cuando el resultado devuelto por un método no es igual al valor correcto

**Error:**

Ocurre cuando el programa compila correctamente, pero al ejecutarse ocurre un error.

1. **Las implementaciones de deberiaPasar, deberiaFallar, deberiaErrar**

**se encuentran en bluej.**

**Prácticando Pruebas en BlueJ [En lab02.doc \*.java]**

**De TDD → BDD (TEST → BEHAVIOUR DRIVEN DEVELOPMENT)**

**Ahora vamos escribir el código necesario para que las pruebas de FraccionarioTest.**

**1.Determinen las estructuras de datos necesarias para almacenar los elementos de un fraccionario. Justifique la selección.**

Se puede usar una lista de tamaño 2, donde la posición 0 representa el numerador y la 1

El denominador.

Para nuestro caso no la usamos ya que, se puede modelar a través de atributos del objeto, es decir usamos dos atributos llamados numerador y denominador.

**2. Implementen los métodos necesarios para pasar todas las pruebas definidas. ¿Cuáles métodos implementaron?**

No implementamos métodos adicionales ya que los que

**Desarrollando BDD - MDD [En lab02.doc, calmatfra.asta, \*.java]**

Para desarrollar esta aplicación vamos a considerar los siguientes ciclos de desarrollo.

Ciclo 1 : Operaciones de básicas: asigne, consulte

Ciclo 2 : Operaciones aditivas: sume y reste

Ciclo 3 : Operaciones multiplicativas: multiplicación

Ciclo 5 : Proponga dos nuevas funcionalidades

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Ciclo | Calmatfra | CalmatfraTest | Matriz | MatrizTest | Fraccionario | FraccionarioTest |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |

**RETROSPECTIVA**

**1. ¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes?**

**(Horas/Hombre)**

Iván Camilo Rincon 20 horas

Miguel Ángel Fúquene Arias 20 hora

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

Casi terminado, puesto que le invertimos el tiempo y la disposición durante toda la semana.

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

La programación a pares, ya que con estos ambos integrantes del grupo aprenden los temas de forma igualitaria.

**4. ¿Cuál consideran fue el mayor logro? ¿Por qué?**

Pudimos aplicar los conceptos vistos en clase satisfactoriamente puesto que pudimos realizar todas las funcionalidades que se requerían.

**5. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para**

**resolverlo?**

Tuvimos un poco de inconvenientes a la hora de entender lo que pedía el laboratorio en algunos puntos sin embargo lo podemos resolver leyendo nuevamente el laboratorio.

**6. ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los**

**resultados?**

Distribuimos bien el tiempo y utilizamos espacios de estudio fuera de clase para fortalecer los conocimientos y poder realizar todas las funcionalidades que proponía el laboratorio. Nos comprometemos a reforzar los temas vistos.