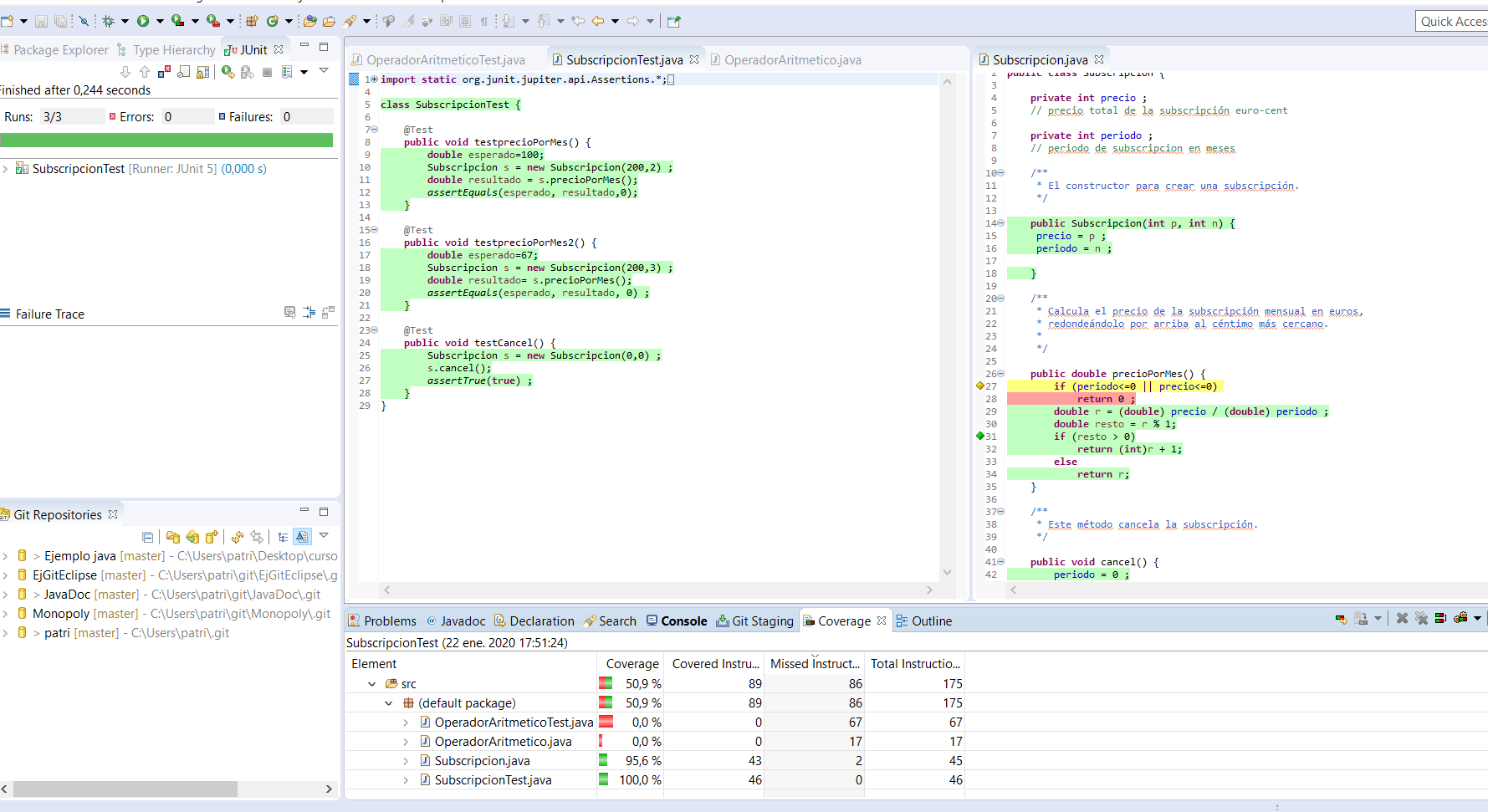
1. He creado un proyecto JUnitLab en la que dentro he metido la clase Suscripción que venía dada en nuestro pdf. El periodo de subscripción se representa en meses y se recoge en la variable periodo.
2. A continuación he creado una clase de prueba en la que aparece llamada por defecto SubscripcionTest, después he creado los casos de prueba donde la clase llamada SubscripcionTest ha sido rellenada con el código que viene también dado en el pdf.
3. He ejecutado la clase de prueba para que nos informe que fallos han tenido las dos clases creadas, en este caso nos informa que esperaba 1.0 y ha recibido como respuesta 100.0 podemos observar que el valor no estaba correctamente definido.
4. Modificamos la variable esperado=100 en el caso de prueba y volvemos a ejecutarlo, y comprobaremos que ahora si se ejecuta correctamente. También añadimos el testCancel a la clase SubscripcionTest de la clase a probar(Suscripcion).

Aquí podremos ver que en el código de la clase a probar (Susbscripcion) tiene una cobertura 100%, por lo que las clases Suscripcion y SubscripcionTest nos han quedado de esta manera:



1. He creado una nueva clase llamada OperadorAritmetico, que realiza las operaciones de suma y división, para probar la clase he creado una clase test llamada OperadorAritmeticoTest que en principio solo pruebe la suma, estas dos clases las rellenamos con los códigos que tenemos en el pdf (testSuma y testDivision).
2. Pulsamos en el botón de EclEmma y seleccionamos la opción coverage as Junit Test, vemos a continuación que la pestaña coverage nos muestra el informe de cobertura del proyecto y en este caso nos da fallos.
3. A continuación creamos un caso de prueba para la división dentro de la clase OperadorAritmeticoTest. Si ahora volvemos a realizar las pruebas, vemos que la situación ha mejorado, aunque todavía queda código sin cubrir.
4. Añadimos a la clase OperadorAritmeticoTest las operaciones de testSumaNegativo, testDivisionPorCero y testConstructor, de esta manera vemos como poco a poco hay menos fallos.
5. Le añadimos al testSuma la variable oa, y a la clase OperadorAritmetico le añadimos el modificador de public. Ejecutamos la clase de prueba y comprobamos que la cobertura está al 100%.

Aquí podremos ver que en el código de la clase a probar (OperadorAritmetico) tiene una cobertura 100%, por lo que las clases OperadorAritmetico y OperadorAritmeticoTest nos han quedado de esta manera:

