

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN
DISEÑO DE SOFTWARE
TALLER DE PRUEBAS UNITARIAS - I TÉRMINO 2023

Objetivos Específicos

1. Crear casos de prueba para la validación y verificación de un producto de software utilizando el marco de trabajo JUnit5.
2. Evaluar el uso de una herramienta informática para automatización de las pruebas unitarias en un producto de software.

Resultado de Aprendizaje

1. Habilidad para aplicar teoría de ciencias computacionales y fundamentos de desarrollo de software para producir soluciones basadas en computación.

Requerimiento

- IDE - IntelliJ IDEA
- Maven

Descripción

En equipos de trabajo, constituido por hasta 4 estudiantes, implemente las pruebas unitarias asociadas a los casos de prueba indicados para asegurar que el código adjunto funciona correctamente. Además, realizar un reporte indicando los datos para probar cada uno de los casos de prueba y realizar una valoración grupal sobre el uso del marco de trabajo.

Especificaciones

Considere la clase adjunta Calculadora.java para realizar las pruebas unitarias correspondientes a los casos de prueba indicados a continuación:

Casos de prueba:

Suma:

- Suma básica
- Suma de números negativos
- Suma de números positivos

Resta:

- Resta básica
- Resta de números negativos
- Resta de números positivos

Multiplicación:

- Multiplicación básica
- Multiplicación de un número negativo con uno positivo
- Multiplicación de dos números negativos
- Multiplicación de dos números positivos
- Multiplicación por cero

División:

- División básica
- División entre cero
- División de número negativo por positivo

Radicación:

- Raíz cuadrada de un numero positivo
- Raíz cuadrada de un número negativo
- Raíz enésima

Exponenciación:

- Potencia positiva
- Potencia con base negativa y exponente par
- Potencia con base negativa y exponente impar
- Potencia de cero

Sección A

Elabore un reporte en el que coloque los datos para probar cada caso de prueba indicado en las especificaciones. Agregar las capturas de imagen correspondientes a la ejecución exitosa de cada prueba unitaria y colocarla en el caso de prueba correspondiente. [20%]

Sección B

Configurar el proyecto de java para que utilice el marco de trabajo Junit5 e implementar las pruebas unitarias asociadas a cada uno de los casos de prueba requeridos. Utilizar `@DisplayName` para indicar el caso de prueba asociado a cada prueba unitaria. Utilizar `@BeforeEach`, `@AfterEach`, `@BeforeAll`, `@AfterAll` para evitar repetir código innecesariamente. Por lo menos utilizar 5 diferentes tipos de métodos Assert, por ejemplo: `assertEquals`, `assertTrue`, `assertNotNull`, `assertAll`, `assertThrows` [50%]

Sección C

Finalmente, evaluar el uso del marco de trabajo, indicando cuales son los beneficios y desventajas sobre su uso dentro del proceso de validación y verificación de un producto de software. [30%]

Recomendaciones

- Aprender a utilizar Maven es importante ya que JUnit se convierte en una dependencia que puede ser administrada eficientemente a través de Maven
- En caso de que no tenga conocimientos de cómo usar JUnit 5 en IntelliJ IDEA mediante maven ver este video: <https://youtu.be/jBeu8BDvM48>
- Consultar la documentación de JUnit 5: <https://junit.org/junit5/docs/current/user-guide/>
- Examinar y entender el contenido completo de este video que proporciona instrucciones desde nivel básico sobre la implementación de aspectos relacionados con las pruebas, incluyendo pruebas unitarias y el uso de Mockito: <https://youtu.be/Geq60OVyBPg>

Entregables (en un archivo ZIP o RAR)

1. Un **documento (disponible en formatos docx y pdf)** que contenga el reporte y las imágenes correspondientes. Además, el documento debe tener una carátula en la que se liste a los integrantes del equipo, una tabla de contenido.
2. Cada estudiante debe entregar una evaluación a los miembros de su equipo de trabajo usando la rúbrica de co-evaluación en la **tarea individual** “Co-evaluación Taller Pruebas Unitarias” a disposición en Aula Virtual.

Rúbrica de Calificación

Descripción	Valor
Casos de prueba en el reporte	20
Implementación de todas las pruebas unitarias	20
Uso de anotaciones para Junit5	10
Uso de al menos 5 tipos de aserciones	20
Evaluación del uso de framework Junit5	30
Total	100
No subir a Aula Virtual los entregables requeridos de acuerdo con lo especificado	-100

Late Submission Policy

Delay (§)	Penalty (Ω)
1 hour or less	loss of 10%
1 to 6 hours	loss of 20%
6 to 24 hours	loss of 30%
Over 24 hours:	loss of 100%

(§) every clock hour counts including weekends or holidays

(Ω) automatic and non-negotiable penalty