ENTORNOS DE DESARROLLO

PRUEBAS DE CAJA BLANCA, CAJA NEGRA E IMPLEMENTACIÓN EN JUNIT

Alumnos: Alberto Olmo Morago

Alberto Martínez Pérez

ÍNDICE

[1. PRUEBAS DE CAJA BLANCA 1](#_Toc135306972)

[A. DIAGRAMA DE GRAFOS 1](#_Toc135306973)

[B. REGIONES 1](#_Toc135306974)

[C. COMPLEJIDAD CICLOMÁTICA 1](#_Toc135306975)

[D. CAMINOS DE PRUEBA 1](#_Toc135306976)

[E. TABLA CON CASOS DE PRUEBA PARA CADA CAMINO 2](#_Toc135306977)

[2. PRUEBAS DE CAJA NEGRA 2](#_Toc135306978)

[A. TABLA DE CLASES DE EQUIVALENCIA 2](#_Toc135306979)

[B. TABLA DE ANÁLISIS DE VALORES LÍMITE 2](#_Toc135306980)

[C. CASOS DE PRUEBA 2](#_Toc135306981)

[3. PRUEBAS DE JUNIT 3](#_Toc135306982)

[4. DEPURACIÓN 3](#_Toc135306983)

# PRUEBAS DE CAJA BLANCA

## DIAGRAMA DE GRAFOS

Un dibujo de una persona

Descripción generada automáticamente con confianza media

NOTA: El nodo 4 nunca podría llegar a ejecutarse como true porque de introducir -3, el nodo 1 sería true siempre.

## REGIONES

Tiene 3 regiones.

## COMPLEJIDAD CICLOMÁTICA

* Regiones: 3.
* Aristas-Nodos: 10 – 9 + 2 = 3.
* Nodos de condición: 2 + 1 = 3.

## CAMINOS DE PRUEBA

* Camino 1: Inicio à 1 à 2 à 3 à FIN.
* Camino 2: Inicio à 1 à 4 à 5 à FIN.
* Camino 3: Inicio à 1 à 4 à 6 à 7 à FIN.

## TABLA CON CASOS DE PRUEBA PARA CADA CAMINO

|  |  |  |
| --- | --- | --- |
| Camino | Cantidad | Resultado |
| 1 | -1 | 1 |
| 2 | -3 | 1 |
| 3 | 1 | 0 |

# PRUEBAS DE CAJA NEGRA

## TABLA DE CLASES DE EQUIVALENCIA

|  |  |  |
| --- | --- | --- |
| Condición a analizar | Clase válida | Clase no válida |
| Cantidad positiva | (1) 20 | (2) -100 |
| Cantidad menor al saldo | (3) Cantidad < Saldo | (4) Cantidad > Saldo |

## TABLA DE ANÁLISIS DE VALORES LÍMITE

|  |  |  |
| --- | --- | --- |
| Condición a analizar | Clase válida | Clase no válida |
| Rango positivo | (5) 0 | (6) -1 |
| Cantidad menor al saldo | (7) Saldo | (8) Saldo + 1 |

## CASOS DE PRUEBA

|  |  |  |  |
| --- | --- | --- | --- |
| Caso | Clases válidas | Clase no válida | Resultado |
| 0 | (1, 3, 5, 7) | - | “Se retira n dinero” |
| -1 | - | (2, 6) | “No se puede retirar una cantidad negativa. |
| Saldo + 1 | - | (8) | “No se puede retirar una cantidad mayor al saldo” |

# PRUEBAS DE JUNIT

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Se modifica el orden para que al ingresar -3 produzca error.

# DEPURACIÓN

* Punto de parada sin condición al crear el objeto miCuenta en la función main. Línea 3 del código del método main que se presenta en la siguiente página de este libro.

Captura de pantalla de un celular

Descripción generada automáticamente

* Punto de parada en la instrucción return del método ingresar sólo si la cantidad a ingresar es menor de 0. Línea 20 del código del método ingresar que se presenta más adelante.

Captura de pantalla de un celular

Descripción generada automáticamente

* Punto de parada en la instrucción donde se actualiza el saldo, sólo deberá parar la tercera vez que sea actualizado. Línea 16 del código del método ingresar que se presenta más adelante.

Captura de pantalla de un celular

Descripción generada automáticamente

NOTA: El “pass count” de IntelliJ es lo mismo que “hit count” en Eclipse.