Nomes: Arthur Pretto, Brenda David, Juliano Terra, Victoria Fraga

Particionamento:

Partição	Casos de Testes	Resultado Esperado
CategoriaSilver	49.000	49.000, SILVER
CategoriaSilverParametrizado	49.000, 10, SILVER	49.000, 10, SILVER
CategoriaGold	50.000	50.000, GOLD
CategoriaGoldParametrizada	55.000, 55.000, GOLD	55.000, 55.000, GOLD
CategoriaMudancaPlatinum	50.000,200.000	252.000, PLATINUM
CategoriaPlatinum	50.000, 200.000, 1.000	253.025, PLATINUM

Valores limites:

SILVER

on-points = 50_000

off-points= 49_999

in points = 1_000, 10_0000, 30_0000

out points = 60_000, 70_000, 100_000

GOLD

on-points = 200_000

off-points= 199_999

in points = 100_000, 150_0000, 190_0000

out points = 200_000, 250_000, 350_000

PLATINUM

on-points = 199_999

off-points= 200_000

in points = 200_000, 250_0000, 300_0000

out points = 50_000, 100_000, 150_000

Após rodar cenários iniciais, atingimos 95% de abrangência nos testes.

default

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
ContaCorrente		95%		75%	8	19	2	33	2	7	0	1
CategoriaEnum		100%		n/a	0	1	0	4	0	1	0	1
Total	6 of 177	96%	6 of 24	75%	8	20	2	37	2	8	0	2

Os seguintes casos de teste foram adicionados:

```
@Test
public void validaInformacoesConta(){

    assertEquals("1234-3",conta.getNumeroConta());
    assertEquals("Joao Ribeiro",conta.getNomeCorrentista());

}
```

```
@Test
public void depositoMudaCategoriaPlatinumLimite() {
    depositoInicial(58_600);
    double valorDeposito = 140_000;

    boolean resultadoDeposito = conta.deposito(valorDeposito);

    assertEquals(200_000, conta.getSaldo());
    assertEquals(CategoriaEnum.PLATINUM, conta.getCategoria());
}
```

```
@Test
public void retiradaContinuaGold() {
    depositoInicial(50_000);
    double valorSaque = 10_000;

    boolean resultadoSaque = conta.retirada(valorSaque);

    assertTrue(resultadoSaque);
    assertEquals(30_000, conta.getSaldo());
    assertEquals(CategoriaEnum.GOLD, conta.getCategoria());
}
```

```java
@Test
public void retiradaGoldLimite() {
    depositoInicial(50_000);
    double valorSaque = 25_000;

    boolean resultadoSaque = conta.retirada(valorSaque);

    assertTrue(resultadoSaque);
    assertEquals(25_000, conta.getSaldo());
    assertEquals(CategoriaEnum.GOLD, conta.getCategoria());
}
```

Ao final não conseguimos cobrir todos os cenários chegando aos números abaixo:

ContaCorrente

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods
deposito(double)		100%		91%	1	7	0	14	0	1
retirada(double)		100%		90%	1	6	0	9	0	1
ContaCorrente(String, String)		100%		n/a	0	1	0	6	0	1
getNumeroConta()		100%		n/a	0	1	0	1	0	1
getNomeCorrentista()		100%		n/a	0	1	0	1	0	1
getSaldo()		100%		n/a	0	1	0	1	0	1
getCategoria()		100%		n/a	0	1	0	1	0	1
Total	0 of 138	100%	2 of 22	90%	2	18	0	33	0	7