

RAPPORT DE TESTS LOGICIELS-TP2

RÉALISÉ PAR:

Hadil Helali

Soulaima Khala

Raoua Trimech

Mouhamed Zied Brahmi



La classe PolynomeSecondDegre

```
PolynomeSecondDegre.java ×
1 package tp_test2;
2
3 public class PolynomeSecondDegre {
4     private double a, b, c;
5     private double[] racines;
6     public PolynomeSecondDegre(double a, double b, double c) {
7         super();
8         this.a = a;
9         this.b = b;
10        this.c = c;
11    }
12    public PolynomeSecondDegre() {
13    }
14    public double discriminant() {
15        return b * b - 4 * a * c;
16    }
17    public double[] calculRacines() {
18        racines = null;
19        double discr = discriminant();
20        if (discr == 0) {
21            racines = new double[1];
22            racines[0] = -b / (2 * a);
23        } else if (discr > 0) {
24            double racineDiscriminant = Math.sqrt(discr);
25            racines = new double[2];
26            racines[0] = (-b + racineDiscriminant) / (2 * a);
27            racines[1] = (-b - racineDiscriminant) / (2 * a);
28        }
29        return racines;
30    }
31 }
```

Les tests de la classe PolynomeSecondDegre

```
PolynomeSecondDegre.java TestEquationSecondDegre.java ×
1 package tp_test2;
2
3
4 import static org.junit.Assert.assertEquals;
5
6
7
8
9
10
11
12
13
14 public class TestEquationSecondDegre extends TestCase{
15
16     @Test
17     public void test2Racines() {
18         PolynomeSecondDegre p = new PolynomeSecondDegre(1, 1, -2);
19         double[] expectedRacines = {1, -2};
20         assertEquals(expectedRacines, p.calculRacines(), 0);
21     }
22
23     @Test
24     public void test1Racine() {
25         PolynomeSecondDegre p = new PolynomeSecondDegre(1, -2, 1);
26         double[] expectedRacines = {1};
27         assertTrue(Arrays.equals(expectedRacines, p.calculRacines()));
28     }
29
30     @Test
31     public void testPasDeRacineReelle() {
32         PolynomeSecondDegre p = new PolynomeSecondDegre(1, 1, 1);
33         double[] expectedRacines = null;
34         assertEquals(expectedRacines, p.calculRacines());
35     }
36
37
38 }
39
```

L'exécution des tests

Finished after 0,072 seconds

Runs: 3/3 Errors: 0 Failures: 0

TestEquationSecondDegre [Runner: JUnit 5] (0,001 s)

- testPasDeRacineReelle (0,000 s)
- test2Racines (0,001 s)
- test1Racine (0,000 s)

Dans notre cas , **tous les tests** sont exécutés avec **succès**.

Cependant , Si le test échoue, JUnit stockera les informations sur l'erreur ou l'exception dans l'objet **result**, y compris le type d'erreur et son message. Les informations sur l'erreur peuvent être récupérées à partir de l'objet **result** après l'exécution du test.