

```

package player;

/** Represents a simple 3-degree polynomial */
public class SimplePolynomial {
    private int c0, c1, c2, c3;

    public SimplePolynomial(int c3, int c2, int c1, int c0) {
        this.c0 = c0;
        this.c1 = c1;
        this.c2 = c2;
        this.c3 = c3;
    }

    /** Solves this polynomial for the given value */
    public double solve(double x) {
        return (((((c3 * x) + c2) * x) + c1) * x) + c0;
    }

    /** Gets the derivative of this polynomial. */
    public SimplePolynomial getDerivative() {
        return new SimplePolynomial(0, 3 * c3, 2 * c2, c1);
    }

    /** Gets the zeros of a 2-degree polynomial */
    public double[] getZeros() {
        if (c3 != 0)
            throw new RuntimeException("Cannot find the zeros of a 3-degree polynomial");

        if (c2 == 0) {
            if (c1 == 0)
                return new double[0];
            // Linear equation
            return new double[] { -c0 / (double) c1 };
        }

        // Quadratic
        double discriminant = (c1 * c1) - (4.0 * c2 * c0);
        if (discriminant < 0) {
            return new double[0];
        }
        else if (discriminant == 0)
            return new double[] { -c1 / (2.0 * c2) };
        else
        {
            double sqrt_d = Math.sqrt(discriminant);
            return new double[] { (-c1 + sqrt_d) / (2.0 * c2),
                                   (-c1 - sqrt_d) / (2.0 * c2) };
        }
    }
}

```