AP Problem: APParabola

Part 1:

An `APParabola` is a parabola defined by the equation $ax^2 + bx + c = 0$, where *a*, *b*, and *c* are integers and *a* is not zero. The zeros of the parabola are defined to be the double values $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$ and $\frac{-b - \sqrt{b^2 - 4ac}}{2a}$. An `APParabola` opens up if the value of *a* is positive, and opens down if the value of a is negative. Examples of three `APParabola` equations are shown in the following table.

Equation	**Zeros**	**Does the parabola open up or down?**
$1x^2 + x - 2 = 0$	$x_1 = -2 \quad x_2 = 1$	Up, because *a* is positive
$-x^2 + 4x - 4 = 0$	$x_1 = 2 \quad x_2 = 2$	Down, because *a* is negative
$x^2 + 0x + 2 = 0$	Undefined – no zeros	Up, because *a* is positive

Assume that the following code segment appears in a class other than `APParabola`. The code segment shows an example of using the `APParabola` class to represent two equations shown in the table.

```
APParabola parabola1 = new APParabola(1, 1, -2);
double p1Zero1 = parabola1.getZero1( );
double p1Zero2 = parabola1.getZero2( );
String openDir1 = parabola1.Opens( );

APParabola parabola2 = new APParabola(-1, 4, -4);
double p2Zero1 = parabola1.getZero1( );
double p2Zero2 = parabola1.getZero2( );
String openDir2 = parabola1.Opens( );
```

Write the `APParabola` class. Your implementation must include a constructor that has three parameters that represent *a, b,* and *c,* in that order. You may assume that the value of the parameter representing *a* is not zero. It must also include a method `getZero1` that calculates one zero using $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$ and returns the value, and a method `getZero2` that calculates one zero using $\frac{-b - \sqrt{b^2 - 4ac}}{2a}$ and returns the value. The class must also contain a method `Opens` that returns "up" if the value of *a* is positive and returns "down" otherwise. Your class must produce the indicated results when invoked by the code segment given above.

Part 2:

Write a driver class that prints the values of the zeros and the direction that the parabola opens for each of the examples in part 1. The code for creating the object and calling the methods is given to you in part one.