Laboratory Assignment № 2

Dr. Duncan, CSC 1350, Louisiana State University 09/17/2019

Characteristics of a Parabola

Learning Objectives

1. Performing Arithmetic Operations and Using Standard Java Math Library Methods
2. Using the Formatted Print Statement

In today's laboratory session, you will write a program that computes various quantities that are used in characterizing a parabola. A quadratic function is a real-valued second order polynomial function in the form $y = f(x) = ax^2 + bx + c$, where a, b and c are real numbers. The curve that describes a quadratic function is called a **parabola**.

DEFINITION 1. A **Quadratic Equation** is a second-order polynomial equation in a single variable x given by the expression below.

$$ax^2 + bx + c = 0 \tag{1}$$

with $a \neq 0$. a is referred to as the coefficient of the quadratic term, b, the coefficient of the linear term, and c, the constant term. Because a quadratic equation is a second-order polynomial equation, the fundamental theorem of algebra guarantees that it has two solutions.

DEFINITION 2. The quantity $\mathbb{D} = b^2 - 4ac$ is called the **discriminant** of a quadratic equation.

DEFINITION 3. The **axis of symmetry** of a parabola is a vertical line that divides the parabola into two congruent halves. Given any horizontal line that intersects the parabola at two points, both points are equidistant from the axis of symmetry. The axis of symmetry of a parabola is the vertical line $x = -\frac{b}{2a}$.

DEFINITION 4. A **vertex** of a parabola is the point at which it crosses the axis of symmetry. It is the lowest point (*minimum*) when the parabola is concave upward and the highest point (*maximum*) when it is concave downward. The vertex of a parabola is the point $\left(\frac{-b}{2a}, \frac{-b^2}{4a} + c\right)$

DEFINITION 5. The **x-intercepts** are the points at which the parabola crosses the x-axis; that is, the points (x, y) such that y is 0. The x-coordinates of the x-intercepts are the roots of the equation. A quadratic equation may have zero, one or two intercepts. The **y-intercept** is the point at which the parabola crosses the y-axis; that is, the point (x, y) such that x is 0. The x-intercepts, when they exists, are $\left(\frac{-b-\sqrt{\mathbb{D}}}{2a}, 0\right)$ and $\left(\frac{-b+\sqrt{\mathbb{D}}}{2a}, 0\right)$, where $\mathbb{D}$ is the discriminant of the quadratic equation. The y-intercept is $(0, c)$, where c is the constant term. When the discriminant is negative, the parabola does not intersect the x-axis. The square root of a negative number is a complex number. In Java, when the square root of a negative number is calculated, the Java library constant **NaN** is returned. *NaN* is the acronym for **N**ot **a N**umber.

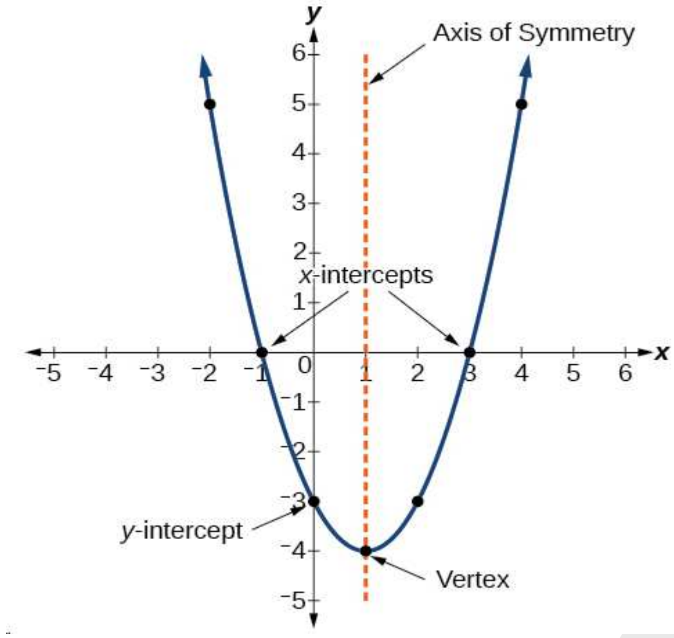

Figure 1: A Parabola for $y = x^2 - 2x - 3$ with Some of its Characteristics

The solutions to a quadratic equation, also referred to as its roots, are the x-coordinates of the x-intercepts, if they are real numbers. The roots are real numbers if the parabola intersects the x-axis. If the x-coordinates of the x-intercepts are complex numbers (NaN), then the roots are also complex numbers. In this lab exercise, your program will output NaN whenever a root or intercept is a complex number.

The *QuadraticSolver* Program

Write a Java program that prompts the user for the coefficient of the quadratic term, the coefficient of the linear term, and the constant term of a quadratic equation. The program then computes and displays its discriminant, roots, axis of symmetry, vertex, x-intercepts and y-intercept. Assume that when the user is prompted to enter the coefficient of the quadratic term, a nonzero number is entered. Your program is not required to check whether the user does so.

Listing 1: Sample Run

```
To solve the quadratic equation, aX^2 + bx + c = 0,
enter the non-zero coefficient of its quadratic term, a, the
coefficient of its linear term, b, and its constant term, c:
1 -6 9

Discriminant: 0.00
Axis of Symmetry: x = 3.00
Vertex: (3.00, 0.00)
y-intercept: (0.00, 9.00)
x-intercepts: (3.00, 0.00) and (3.00, 0.00)
roots: {3.00, 3.00}
```

Listing 2: Sample Run

```
To solve the quadratic equation, aX^2 + bx + c = 0,
enter the non-zero coefficient of its quadratic term, a, the
coefficient of its linear term, b, and its constant term, c:
-4 0 64

Discriminant: 1024.00
Axis of Symmetry: x = 0.00
Vertex: (0.00, 64.00)
y-intercept: (0.00, 64.00)
x-intercepts: (4.00, 0.00) and (-4.00, 0.00)
roots: {4.00, -4.00} (4.00, 0.00) and (-4.00, 0.00)
Root: x = {4.00, -4.00}
```

Listing 3: Sample Run

```
To solve the quadratic equation, aX^2 + bx + c = 0,
enter the non-zero coefficient of its quadratic term, a, the
coefficient of its linear term, b, and its constant term, c:
3 45 0

Discriminant: 2025.00
Axis of Symmetry: x = -7.50
Vertex: (-7.50, -168.75)
y-intercept: (0.00, 0.00)
x-intercepts: (-15.00, 0.00) and (0.00, 0.00)
roots: {-15.00, 0.00}
```

Listing 4: Sample Run

```
To solve the quadratic equation, aX^2 + bx + c = 0,
enter the non-zero coefficient of its quadratic term, a, the
coefficient of its linear term, b, and its constant term, c:
12 -7 -12

Discriminant: 625.00
Axis of Symmetry: x = 0.29
Vertex: (0.29, -13.02)
y-intercept: (0.00, -12.00)
x-intercepts: (-0.75, 0.00) and (1.33, 0.00)
roots: {-0.75, 1.33}
```

Listing 5: Sample Run

```
To solve the quadratic equation, aX^2 + bx + c = 0,
enter the non-zero coefficient of its quadratic term, a, the
coefficient of its linear term, b, and its constant term, c:
9 0 16

Discriminant: -576.00
Axis of Symmetry: x = -0.00
Vertex: (-0.00, 16.00)
y-intercept: (0.00, 16.00)
x-intercepts: (NaN, 0.00) and (NaN, 0.00)
roots: {NaN, NaN}
```

Listing 6: Sample Run

```
To solve the quadratic equation, aX^2 + bx + c = 0,
enter the non-zero coefficient of its quadratic term, a, the
coefficient of its linear term, b, and its constant term, c:
4 -12 25

Discriminant: -256.00
Axis of Symmetry: x = 1.50
Vertex: (1.50, 16.00)
y-intercept: (0.00, 25.00)
x-intercepts: (NaN, 0.00) and (NaN, 0.00)
roots: {NaN, NaN}
```

Compiling and Running The Program

Make sure that your program has the Javadoc below after the package statement and before the definition of the *QuadraticSolver* class.

```
/**
 * Describe what your program does.
 * Course: CS1350 <br>
 * Section: type your section number <br>
 * PAWS ID: type your PAWS ID <br>
 * Lab #: 2 <br>
 * Instructor: Dr. Duncan <br>
 * @author type your name
 * @since the date the program was written
 */
```

Avoid unnecessary computation in your code: do not calculate the same quantity multiple times. Save the calculated value to a variable and retrieve it when needed. Also, if a calculated value is used only once, do not save it to a variable. Use the expression that computes the value in the print statement to display the value.

Although not required, you could use only two print statements to generate the required output: a 'println' statement for the prompt and a 'printf' statement for the output. If you do, a series of successive concatenations will be required.

Select *Run* | *Run Main Project* from the Netbeans menu or, alternatively, click the green ▷ button in the menu bar. Make sure that the output of the program is formatted as shown in the sample runs with numeric outputs displayed to the nearest hundredths.

Archiving and Submitting Your Program

Using windows explorer, navigate your way to the "My Document" folder. Navigate your way through the following path: *NetBeansProjects* | *QuadraticSolver* | *src* | *quadraticsolver*. You will see a file called *QuadraticSolver.java*, your source code. Right-click the source file and choose *Send to* | *Compressed (zipped) folder* from the pop-up menu. Rename the zip file *PAWSID_lab02.zip*, where *PAWSID* is the prefix of your LSU/Tiger email address - the characters left of the @ sign. Drag this zip file to your desktop, verify that it contains your source file and upload it to the drop box on Moodle. You may submit your work as many times as you wish once the due time has not elapsed. You will receive no points if your work is not uploaded to the drop box before the cut-off time.