CS 1073

FR04A

Assignment 5

Ebrahim Arefi

3621326

I. Making a Decision:

```
/**
 * A driver class for As5Q1
 * Represents a chat bot that asks the user questions about their program and GPA.
 * @author Ebrahim Arefi, 3621326
 */

import java.util.Scanner;

public class As5Q1 {
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println();
        System.out.println("-------------------------------------------------------------------
");
        System.out.println("Are you a current undergraduate student at UNB?");

        String undergraduate = sc.nextLine();

        if (undergraduate.equalsIgnoreCase("no")) {
            System.out.println("Sorry, you are not eligible to join");

        } else if (undergraduate.equalsIgnoreCase("yes")) {
            System.out.println("Have you successfuly completed cs 1083 or cs 1023?");
            String csPrereq = sc.nextLine();

            if (csPrereq.equalsIgnoreCase("no")) {
                System.out.println("Sorry, you are not eligible to join");

            } else if (csPrereq.equalsIgnoreCase("yes")) {
                System.out.println("What is your program of study?");
                String programAnswer = sc.nextLine();

                if ((programAnswer.equalsIgnoreCase("BCS")) ||
(programAnswer.equalsIgnoreCase("BSSWE"))
                        || (programAnswer.equalsIgnoreCase("BABCS")) ||
(programAnswer.equalsIgnoreCase("BCSBSC"))) {
                    System.out.println("What is your cumulative GPA?");

                    double gpa = sc.nextDouble();
                    if (gpa >= 2.5) {
                        System.out.println("You may join welcome!");

                    } else {
                        System.out.println("Sorry, you are not eligible to join");
                    }
                } else {
                    System.out.println("Have you participated in a programming contest at UNB?");
                    String contest = sc.nextLine();

                    if (contest.equalsIgnoreCase("no")) {
                        System.out.println("Sorry, you are not eligible to join");
                    } else if (contest.equalsIgnoreCase("yes")) {
                        System.out.println("What is your GPA from CS/SWE classes?");

                        double gpa = sc.nextDouble();
                        if (gpa >= 3.0) {
                            System.out.println("You may join welcome!");
                        }

                    } else {
                        System.out.println("Sorry, you are not eligible to join");
                    }

                }
            }

        }

    }

}
```

Making a Decision Output:

```
ebi@A35-di-Mell As5 % javac As5Q1.java
ebi@A35-di-Mell As5 % java As5Q1

-------------------------------------------------------------------
Are you a current undergraduate student at UNB?
Yes
Have you successfuly completed cs 1083 or cs 1023?
no
Sorry, you are not eligible to join
ebi@A35-di-Mell As5 % java As5Q1

-------------------------------------------------------------------
Are you a current undergraduate student at UNB?
yes
Have you successfuly completed cs 1083 or cs 1023?
yes
What is your program of study?
BCS
What is your cumulative GPA?
3
You may join welcome!
ebi@A35-di-Mell As5 % java As5Q1

-------------------------------------------------------------------
Are you a current undergraduate student at UNB?
yes
Have you successfuly completed cs 1083 or cs 1023?
yes
What is your program of study?
Art
Have you participated in a programming contest at UNB?
no
Sorry, you are not eligible to join
ebi@A35-di-Mell As5 % java As5Q1

-------------------------------------------------------------------
Are you a current undergraduate student at UNB?
yes
Have you successfuly completed cs 1083 or cs 1023?
yes
What is your program of study?
BCSBSC
What is your cumulative GPA?
1
Sorry, you are not eligible to join
```

II. Points and Line Segments

The source code for your `LineSegment` class (Question II)

```
/**
 * This class represents a line segment.
 * Each line has two end points represented by CartesianPoint objects.
 *
 * @author Ebrahim Arefi, 3621326
 */

public class LineSegment {

    /**
     * The first end point of the line segment.
     */
    private CartesianPoint pointA;

    /**
     * The second end point of the line segment.
     */
    private CartesianPoint pointB;

    /**
     * Constructs a LineSegment using two CartesianPoint objects.
     *
     * @param pointA the first endpoint of the line segment.
     * @param pointB the second endpoint of the line segment.
     */
    public LineSegment(CartesianPoint pointAln, CartesianPoint pointBln) {
        pointA = pointAln;
        pointB = pointBln;
    }

    /**
     * Constructs a LineSegment using four coordinate values.
     *
     * @param x1 x-coordinate of the first point.
     * @param y1 y-coordinate of the first point.
     * @param x2 x-coordinate of the second point.
     * @param y2 y-coordinate of the second point.
     */
    public LineSegment(double x1, double y1, double x2, double y2) {
        pointA = new CartesianPoint(x1, y1);
        pointB = new CartesianPoint(x2, y2);

    }

    /**
     * Returns a text description of both end points of the line segment.
```

```java
     *
     * @return a string listing the coordinates of both points.
     */
    public String toString() {
        return "End points: " + "\n\tPoint A: " + pointA + "\n\tpoint B: " + pointB;
    }

    /**
     * Calculates and returns the length of the line segment.
     *
     * @return the distance between the two end points.
     */
    public double length() {
        return pointA.distance(pointB);
    }

    /**
     * Determines whether the line segment crosses either the x-axis or the y-axis.
     *
     * @return true if the line segment crosses an axis, false otherwise.
     */
    public boolean crossesAxis() {
        return (pointA.getY() * pointB.getY() <= 0) || (pointA.getX() * pointB.getX() <= 0);
    }

    /**
     * Determines whether a given CartesianPoint lies on this line segment.
     *
     * @param p the point to be tested.
     * @return true if the point lies on the line segment, false otherwise.
     */
    public boolean containsPoint(CartesianPoint p) {
        double TOLERANCE = 0.00001;

        double ap = pointA.distance(p);
        double pb = p.distance(pointB);
        double ab = pointA.distance(pointB);

        if (Math.abs((ap + pb) - ab) < TOLERANCE) {
            return true;
        } else {
            return false;
        }
    }
}
```

The source code for your `LineSegmentTest` class (Question II)

```
/**
 * A driver class for testing the LineSegment.java
 * It creates and tests 3 line objects.
 *
 * @author Ebrahim Arefi, 3621326
 */

public class LineSegmentTest {
    public static void main(String[] args) {

        CartesianPoint a = new CartesianPoint(2, -3);
        CartesianPoint b = new CartesianPoint(5, 4);

        LineSegment line1 = new LineSegment(a, b);
        LineSegment line2 = new LineSegment(-4, 2, 4, 2);
        LineSegment line3 = new LineSegment(1, 1, 4, 4);

        System.out.println();
        System.out.println("--------------------------");
        System.out.println("ToStrings: ");
        System.out.println();
        System.out.println("line1: " + line1.toString());
        System.out.println("line2: " + line2.toString());
        System.out.println("line3: " + line3.toString());
        System.out.println();

        System.out.println("--------------------------");
        System.out.println("Lengths: ");
        System.out.println();
        System.out.println("Length of line1 is: " + line1.length());
        System.out.println("Length of line2 is: " + line2.length());
        System.out.println("Length of line3 is: " + line3.length());
        System.out.println();

        System.out.println("--------------------------");
        System.out.println("Does it cross?");
        System.out.println();
        if (line1.crossesAxis()) {
            System.out.println("Line 1 crosses at least one axis.");
        } else {
            System.out.println("Line 1 does not cross either axis.");
        }

        if (line2.crossesAxis()) {
            System.out.println("Line 2 crosses at least one axis.");
        } else {
            System.out.println("Line 2 does not cross either axis.");
        }

        if (line3.crossesAxis()) {
            System.out.println("Line 3 crosses at least one axis.");
        } else {
            System.out.println("Line 3 does not cross either axis.");
        }
        System.out.println();

        System.out.println("--------------------------");
        System.out.println("Testings:");
        System.out.println();
        CartesianPoint p1 = new CartesianPoint(0, 2);
        CartesianPoint p2 = new CartesianPoint(2, 2.5);

        if (line2.containsPoint(p1)) {
            System.out.println("Point 1 is on line 2.");
        } else {
            System.out.println("Point 1 is not on line 2.");
        }

        if (line2.containsPoint(p2)) {
            System.out.println("Point 2 is on line 2.");
        } else {
            System.out.println("Point 2 is not on line 2.");

        }
        System.out.println();
        System.out.println("--------------------------");
    }

}
```

The sample output for LineSegmentTest (Question II):

```
ebi@A35-di-Mell As5 % java LineSegmentTest

--------------------------
ToStrings:

line1: End points:
      Point A: CartesianPoint[x=2.0, y=-3.0]
      point B: CartesianPoint[x=5.0, y=4.0]
line2: End points:
      Point A: CartesianPoint[x=-4.0, y=2.0]
      point B: CartesianPoint[x=4.0, y=2.0]
line3: End points:
      Point A: CartesianPoint[x=1.0, y=1.0]
      point B: CartesianPoint[x=4.0, y=4.0]

--------------------------
Lengths:

Length of line1 is: 7.615773105863909
Length of line2 is: 8.0
Length of line3 is: 4.242640687119285

--------------------------
Does it cross?

Line 1 crosses at least one axis.
Line 2 crosses at least one axis.
Line 3 does not cross either axis.

--------------------------
Testings:

Point 1 is on line 2.
Point 2 is not on line 2.

--------------------------
ebi@A35-di-Mell As5 %
```