Module:	COS101
Department:	Computer Science
Assignment:	Term 4 Practical 2
Lecturer:	Mr. C. K. Baker
Due date:	20 September 2024, 5 PM
Total:	50 marks

UWC Computer Science

UNIVERSITY *of the* WESTERN CAPE

Instructions

This practical will test your problem-solving ability using Java programming constructs. There are 2 questions in this assignment. Submit a compressed (.zip) file with all your code, your signed declaration of plagiarism and list of references. The submission file should be named **XXYYZZZ. zip** where **XXYYZZZ** corresponds to your student number.

Question 1: Dice Roll Game [25 marks]

Write a Java program to simulate rolling two dice over 50 rounds. The program should print the values of the dice for each roll; and print *how many rolls,* if any, it takes to get each of the following outcomes:

- Boxcars: both dice show 6.
- Snake eyes: both dice show 1.
- Doublets: both dice show the same number.
- Small straight: an increasing sequence (e.g., 1 and 2, or 5 and 6).
- Ace deuces: a sum of 3 (e.g., 1 and 2, or 2 and 1).

Save your program as `` `Question1.java` ``.

Question 2: Points in 2D space [25 marks]

Here is a definition of a Java class for coordinates in 2D geometric space:

```
class Point{
double x-coord;
double y-coord;
}
```

Create a program `` `Point.java` `` and copy the above definition into it.

Extend your program with the following methods:

- loaded constructor
- getter/accessor methods for each instance variable
- setter/mutator methods for each instance variable
- a method, `gradient(Point p)`, that accepts a coordinate `p`, and returns the gradient of the line passing through the current instance and `p` `(round off to 2 decimal places)`
- a method, `distance(Point p)`, that calculates the Euclidean distance between the current instance and `p`. The formula for Euclidean distance is given by:

  $$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

  Round off your answer to 2 decimal places.
- a method, `is_equal(Point p)`, that accepts a coordinate `p` returns whether the current instance is equal to coordinate `p`
- a method, `midpoint(Point p)`, that accepts a coordinate `p` returns the midpoint of the line passing through the current instance and `p.`

Lastly, test your code using the file `` `Test.java` `` provided to you. When executed, it should display the correct output using the methods defined in your `` `Point.java` `` class.

Marking guide

Question 1

	Mark	Max.	Comment
Program structure and organisation		5	
Correctness of dice simulation		5	
Reporting outcomes		10	
Error-free compilation and code efficiency		5	

Question 2

	Mark	Max.	Comment
Class definition and instance variables		1	
Loaded constructor		2	
Getters / accessors		2	
Setters / mutators		2	
Gradient method		4	
Distance method		4	

Equality method		4	
Midpoint method		4	
Testing and output		2	