

# Jinan University

## Java Programming Lab Report

*Major:* Computer Science & Technology

*School:* International School

*Student name* : \_\_\_\_\_  
(p.s. your name on JNU academic system)

*Student number* : \_\_\_\_\_

*Date of Submission (mm-dd-yyyy):* \_\_\_\_\_

*Instructor:* Yuxia Sun

## Table of Content

|            |                        |   |
|------------|------------------------|---|
| LAB 1      | DATE: 2/28/2023.....   | 3 |
| Problem 1. | (1.5).....             | 3 |
| Problem 2. | (1.9).....             | 3 |
| Problem 3. | (1.11)(Optional).....  | 4 |
| Problem 4. | (2.5).....             | 6 |
| Problem 5. | (2.17).....            | 7 |
| Problem 6. | (2.19) (Optional)..... | 9 |

**LAB 1**      **DATE: 2/28/2023**

**Student Name:**\_\_\_\_\_ **Student ID:** \_\_\_\_\_

**Problem 1. (1.5)**

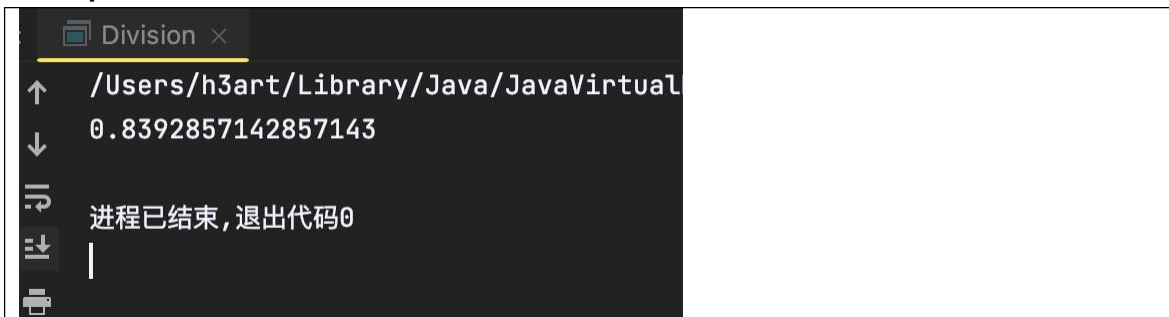
**1.5** (*Compute expressions*) Write a program that displays the result of

$$\frac{9.5 \times 4.5 - 2.5 \times 3}{45.5 - 3.5}.$$

**\* Source Code / Solution :**

```
public class Division {  
    public static void main(String[] args) {  
        double result = (9.5 * 4.5 - 2.5 * 3) / (45.5 - 3.5);  
        System.out.println(result);  
    }  
}
```

**\* Output:**



The screenshot shows a terminal window titled "Division x" with the following output:

```
↑ /Users/h3art/Library/Java/JavaVirtual  
↓ 0.8392857142857143  
⌵ 进程已结束,退出代码0  
⌵ |
```

**\* Debugging/Testing:**

**Bug1:** Forgetting to add parentheses caused an error in the result of the operation.

**Fix:** Add parentheses to the numerator and denominator of the result expression.

**Problem 2. (1.9)**

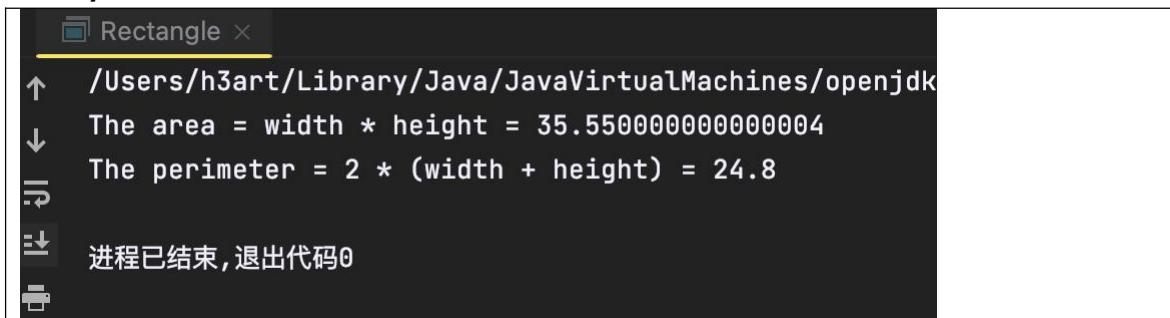
**1.9** (*Area and perimeter of a rectangle*) Write a program that displays the area and perimeter of a rectangle with a width of **4.5** and a height of **7.9** using the following formula:

$$area = width \times height$$

**\* Source Code / Solution :**

```
public class Rectangle {  
    public final double width = 4.5;  
    public final double height = 7.9;  
  
    public static void main(String[] args) {  
        Rectangle obj = new Rectangle();  
        System.out.println("The area = width * height = " + obj.width * obj.height);  
        System.out.println("The perimeter = 2 * (width + height) = " + 2 * (obj.width +  
obj.height));  
    }  
}
```

**\* Output:**

The screenshot shows a terminal window titled "Rectangle x". It displays the output of the Java program: the area calculation "The area = width \* height = 35.550000000000004" and the perimeter calculation "The perimeter = 2 \* (width + height) = 24.8". At the bottom, it shows the message "进程已结束,退出代码0" (Process ended, exit code 0). The terminal has a dark background with light-colored text.

**\* Debugging/Testing:**

**Bug1:** The compiler prompts that a nonstatic variable cannot be referenced from a static context. The program has failed to compile.

**Fix:** We first create a Rectangle instance object and pass it to the static main method. It allows me to use non-static variables in a static context.

**Problem 3. (1.11)(Optional)**

**\*1.11** (Population projection) The U.S. Census Bureau projects population based on the following assumptions:

- One birth every 7 seconds
- One death every 13 seconds
- One new immigrant every 45 seconds

Write a program to display the population for each of the next five years. Assume that the current population is 312,032,486, and one year has 365 days. *Hint:* In Java, if two integers perform division, the result is an integer. The fractional part is truncated. For example,  $5 / 4$  is **1** (not **1.25**) and  $10 / 4$  is **2** (not **2.5**). To get an accurate result with the fractional part, one of the values involved in the division must be a number with a decimal point. For example,  $5.0 / 4$  is **1.25** and  $10 / 4.0$  is **2.5**.

**\* Source Code / Solution :**

```
public class Population {  
    private static final double timeOfYear = 365 * 24 * 60 * 60;  
    private static final double birthTime = 7.0;  
    private static final double deathTime = 13.0;  
    private static final double immigTime = 45.0;  
  
    public static void main(String[] args) {  
        double finalPopulation = 312_032_486;  
  
        for (int i = 0; i < 5; i++) {  
            double increaseNum = timeOfYear / birthTime + timeOfYear / immigTime;  
            double decreaseNum = timeOfYear / deathTime;  
            finalPopulation += increaseNum - decreaseNum;  
  
            System.out.println("After " + (i + 1) + " year, the total population is:");  
            System.out.printf("%d\n", (int)finalPopulation);  
        }  
    }  
}
```

**\* Output:**

```
Population x
/Users/h3art/Library/Java/JavaVirtualMachines
After 1 year, the total population is:
314812582
After 2 year, the total population is:
317592679
After 3 year, the total population is:
320372776
After 4 year, the total population is:
323152872
After 5 year, the total population is:
325932969

进程已结束,退出代码0
```

**\* Debugging/Testing:**

**Bug1:** The original number was stored as an integer, which made the calculation wrong.

**Fix:** In an expression, if a double and an integer are involved in an operation, the result will be stored as double. Therefore, I will store all the numbers given in the question as double, and finally cast back to the output of the integer to ensure that the result is correct.

**Problem 4. (2.5)**

**\*2.5** (Financial application: calculate tips) Write a program that reads the subtotal and the gratuity rate, then computes the gratuity and total. For example, if the user enters **10** for subtotal and **15%** for gratuity rate, the program displays **\$1.5** as gratuity and **\$11.5** as total. Here is a sample run:

```
Enter the subtotal and a gratuity rate: 10 15 Enter
The gratuity is $1.5 and total is $11.5
```

**\* Source Code / Solution :**

```
import java.util.Scanner;

public class FinancialCalculator {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

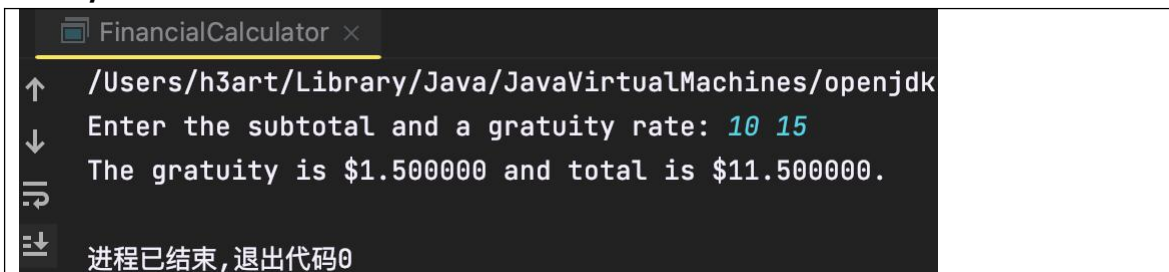
        System.out.print("Enter the subtotal and a gratuity rate: ");
        double subtotal = scanner.nextDouble();
        double gratuityRate = scanner.nextDouble();

        double gratuity = subtotal * gratuityRate * 0.01;
        double total = subtotal + gratuity;

        System.out.printf("The gratuity is $%f and total is $%f.\n", gratuity, total);

    }
}
```

**\* Output:**

A screenshot of a Java IDE terminal window. The window title is "FinancialCalculator x". The terminal shows the following output:   
/Users/h3art/Library/Java/JavaVirtualMachines/openjdk...   
Enter the subtotal and a gratuity rate: 10 15   
The gratuity is \$1.500000 and total is \$11.500000.   
At the bottom, it says "进程已结束,退出代码0" (Process ended, exit code 0).

**\* Debugging/Testing:**

**Bug1:** The wrong form was used to match the input of the Scanner class.

**Fix:** Match the corresponding method of the Scanner instantiated object to the corresponding data type.

**Problem 5. (2.17)**

**\*2.17** (*Science: wind-chill temperature*) How cold is it outside? The temperature alone is not enough to provide the answer. Other factors including wind speed, relative humidity, and sunshine play important roles in determining coldness outside. In 2001, the National Weather Service (NWS) implemented the new wind-chill temperature to measure the coldness using temperature and wind speed. The formula is

$$t_{wc} = 35.74 + 0.6215t_a - 35.75v^{0.16} + 0.4275t_av^{0.16}$$

where  $t_a$  is the outside temperature measured in degrees Fahrenheit,  $v$  is the speed measured in miles per hour, and  $t_{wc}$  is the wind-chill temperature. The formula cannot be used for wind speeds below 2 mph or temperatures below  $-58^{\circ}\text{F}$  or above  $41^{\circ}\text{F}$ .

Write a program that prompts the user to enter a temperature between  $-58^{\circ}\text{F}$  and  $41^{\circ}\text{F}$  and a wind speed greater than or equal to **2** then displays the wind-chill temperature. Use **Math.pow(a, b)** to compute  $v^{0.16}$ . Here is a sample run:

```
Enter the temperature in Fahrenheit between -58°F and 41°F:
5.3 Enter
Enter the wind speed (>= 2) in miles per hour: 6 Enter
The wind chill index is -5.56707
```

**\* Source Code / Solution :**

```
import java.util.Scanner;

public class WindChillCalculator {
    public static double calculator(double ta, double v) {
        return 35.74 + 0.6215 * ta - 35.75 * Math.pow(v, 0.16) + 0.4275 * ta * Math.pow(v, 0.16);
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Enter the temperature in Fahrenheit between -58°F and 41°F: ");
        double outsideTemperature = scanner.nextDouble();
        if (outsideTemperature < -58.0 || outsideTemperature > 41.0) {
            throw new RuntimeException("Temperature is out of range");
        }

        System.out.print("Enter the wind speed (≥ 2) in miles per hour: ");
        double windSpeed = scanner.nextDouble();
        if (windSpeed < 2.0) {
            throw new RuntimeException("Wind speed is out of range");
        }
    }
}
```



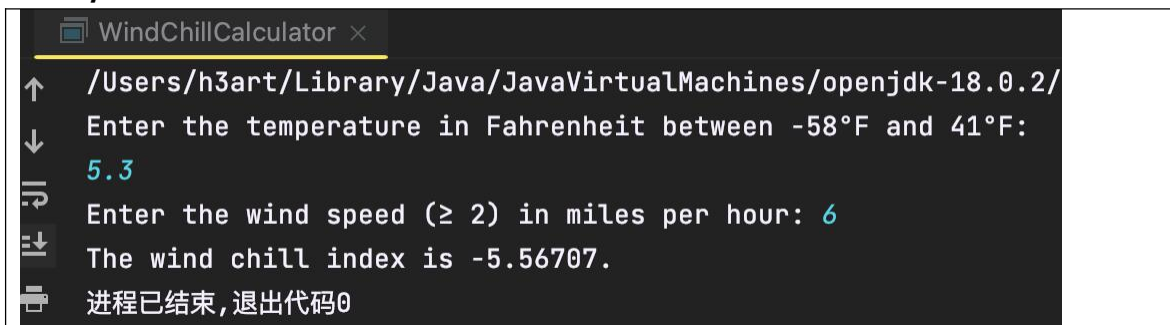
```

    }

    System.out.printf("The wind chill index is %.5f.", calculator(outsideTemperature,
windSpeed));
    }
}

```

**\* Output:**



```

WindChillCalculator x
/Users/h3art/Library/Java/JavaVirtualMachines/openjdk-18.0.2/
Enter the temperature in Fahrenheit between -58°F and 41°F:
5.3
Enter the wind speed (≥ 2) in miles per hour: 6
The wind chill index is -5.56707.
进程已结束,退出代码0

```

**\* Debugging/Testing:**

**Bug1:** The pow () method cannot be called directly.  
**Fix:** The pow() method must use the Math class prefix.  
**Bug2:** Difficult to process incorrect input data.  
**Fix:** The method of throwing an exception is used to make the program stop when it encounters an incorrect data input.

**Problem 6. (2.19) (Optional)**

**\*2.19** (Geometry: area of a triangle) Write a program that prompts the user to enter three points, (x1, y1), (x2, y2), and (x3, y3), of a triangle then displays its area. The formula for computing the area of a triangle is

$$s = (\text{side1} + \text{side2} + \text{side3})/2;$$

$$\text{area} = \sqrt{s(s - \text{side1})(s - \text{side2})(s - \text{side3})}$$

Here is a sample run:

```

Enter the coordinates of three points separated by spaces
like x1 y1 x2 y2 x3 y3: 1.5 -3.4 4.6 5 9.5 -3.4
The area of the triangle is 33.6

```

**\* Source Code / Solution :**

```
import java.util.Scanner;

public class AreaOfTriangle {
    public static double side(double x1, double y1, double x2, double y2) {
        return Math.sqrt(Math.pow(x1 - x2, 2) + Math.pow(y1 - y2, 2));
    }

    public static double area(double[] sides) {
        double factorS = (sides[0] + sides[1] + sides[2]) / 2;
        return Math.sqrt(factorS * (factorS - sides[0]) * (factorS - sides[1]) * (factorS - sides[2]));
    }

    public static boolean sameLineChecker(double[][] points) {
        // Judge  $(y1 - y2)/(x1 - x2) == (y1 - y3)/(x1 - x3) \rightarrow (y1 - y2)(x1 - x3) == (y1 - y3)(x1 - x2)$ 
        return (points[0][1] - points[1][1]) * (points[0][0] - points[2][0])
            == (points[0][1] - points[2][1]) * (points[0][0] - points[1][0]);
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        double[][] points = new double[3][2];
        double[] sides = new double[3];

        System.out.println("Enter the coordinates of three points separated by spaces\n" +
            "like x1 y1 x2 y2 x3 y3:");

        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 2; j++) {
                points[i][j] = scanner.nextDouble();
            }
        }

        if (sameLineChecker(points)) {
            throw new RuntimeException("Not a triangle, points are on the same line");
        }

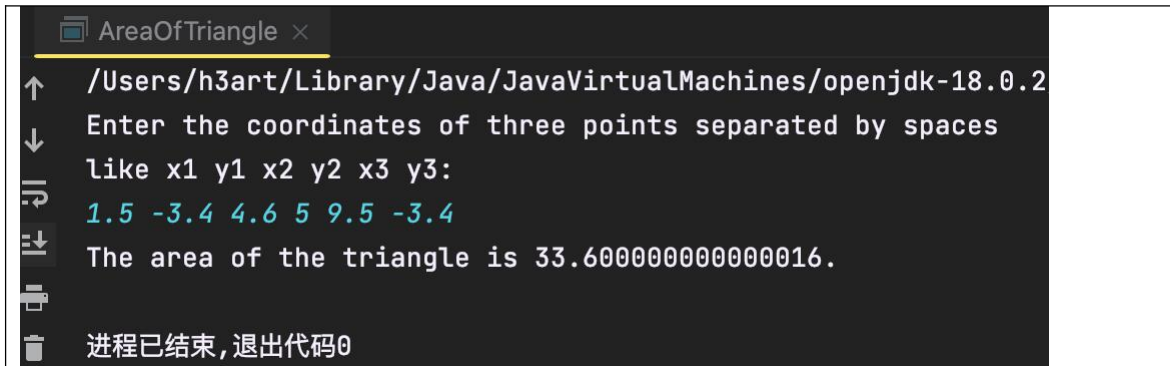
        sides[0] = side(points[0][0], points[0][1], points[1][0], points[1][1]);
        sides[1] = side(points[0][0], points[0][1], points[2][0], points[2][1]);
    }
}
```

```
sides[2] = side(points[1][0], points[1][1], points[2][0], points[2][1]);

System.out.println("The area of the triangle is " + area(sides) + ".");

}
}
```

**\* Output:**



```
AreaOfTriangle x
/Users/h3art/Library/Java/JavaVirtualMachines/openjdk-18.0.2
Enter the coordinates of three points separated by spaces
like x1 y1 x2 y2 x3 y3:
1.5 -3.4 4.6 5 9.5 -3.4
The area of the triangle is 33.600000000000016.
进程已结束,退出代码0
```

**\* Debugging/Testing:**

**Bug1:** I can't define arrays like c/c++.

**Fix:** Use the Java-formatted type `[] = new type [number]` to create the array.

**Bug2:** Forget to check if three points are collinear.

**Fix:** Added a method to determine if three points are collinear. If a triangle cannot be formed, the program throws an exception to terminate.