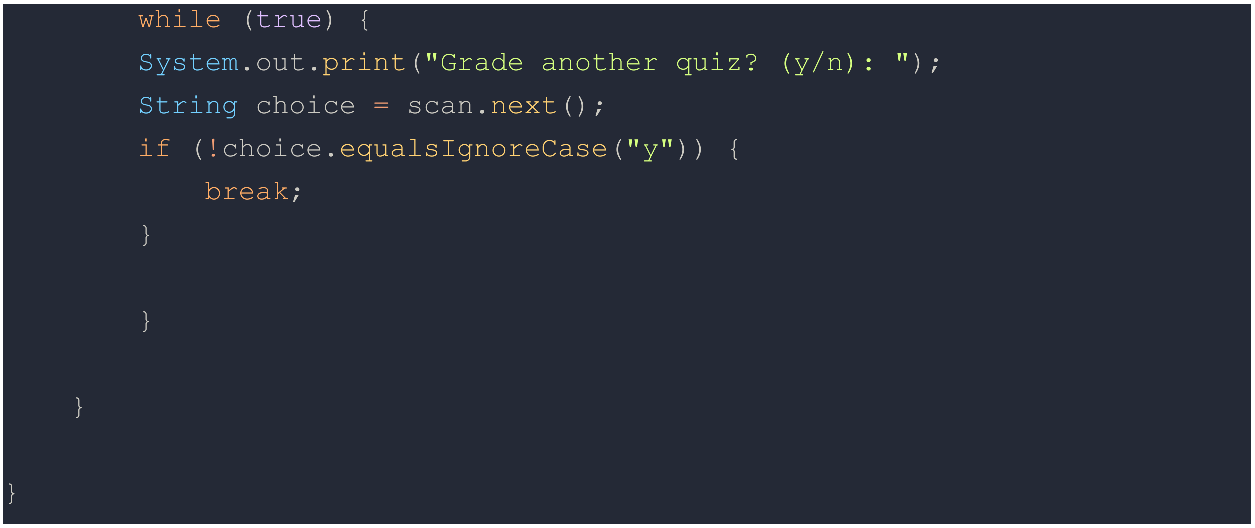
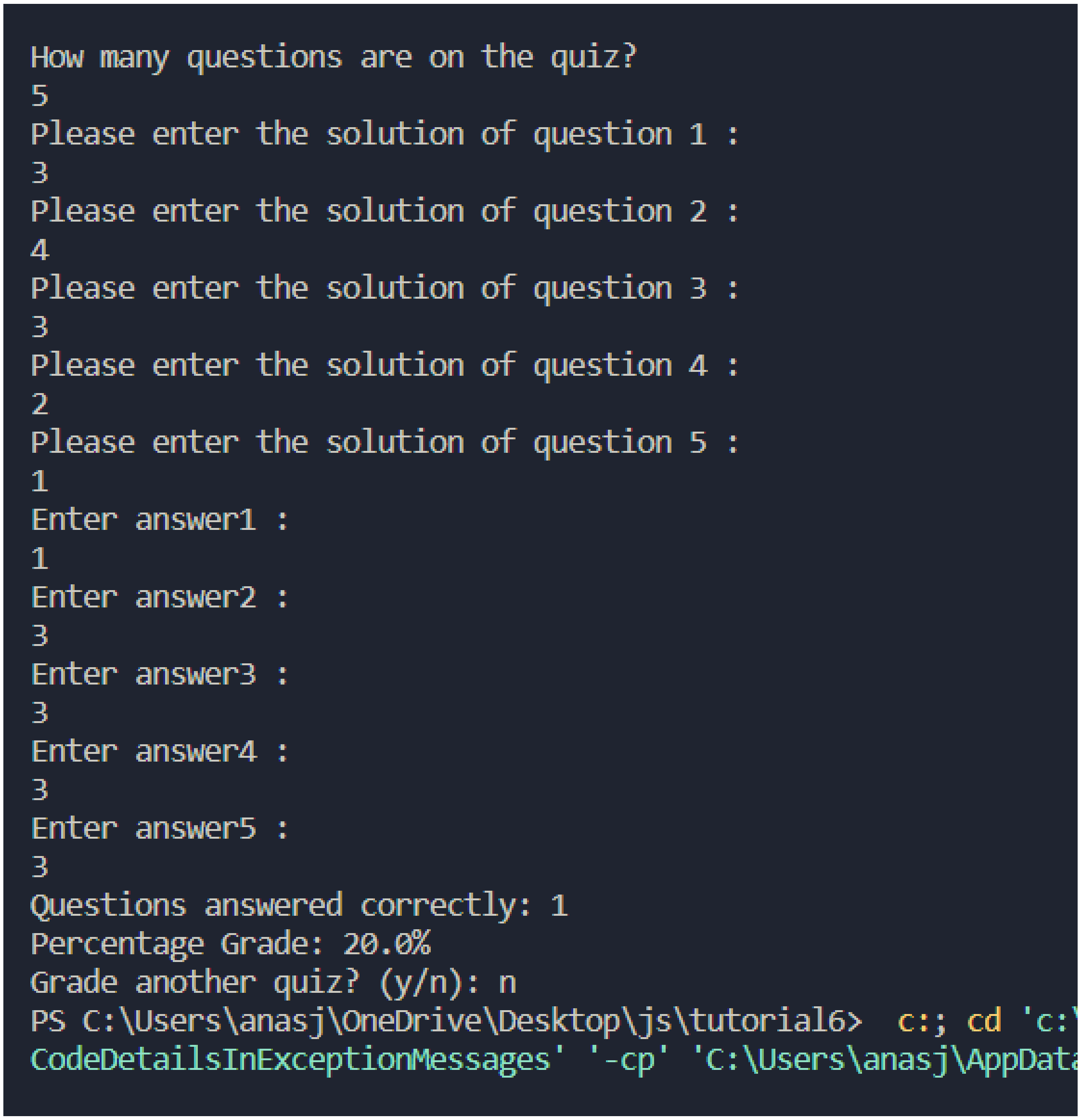
Anas Jamil
100864684
Tutorial # 6 Arrays
SOFE 2710U Object Oriented Programming and Design
FALL 2023

Question 1:

```
import java.util.Scanner;

public class q1 {
    public static void main(String[] args){

        Scanner scan = new Scanner(System.in);
        System.out.println("How many questions are on the quiz? ");
        int n_of_question = scan.nextInt();
        int i;
        int[] key = new int[n_of_question];
        for (i=0; i < n_of_question; i++){

            System.out.println("Please enter the solution of question " +
(i+1) + ": ");
            key[i] = scan.nextInt();
        }
        int[] answers = new int[n_of_question];
        for (int j = 0; j < n_of_question; j++) {

            System.out.println("Enter answer " + (j+1) + ": ");
            answers[j] = scan.nextInt();
        }
        int correct = 0;
        for (int k = 0; k < n_of_question; k++) {
            if (k < key.length && k < answers.length && key[k] ==
answers[k]) {
                correct++;
            }
        }

        double correctp = (double) correct / n_of_question * 100;
        System.out.println("Questions answered correctly: " + correct);
        System.out.println("Percentage Grade: " + correctp + "%");
```



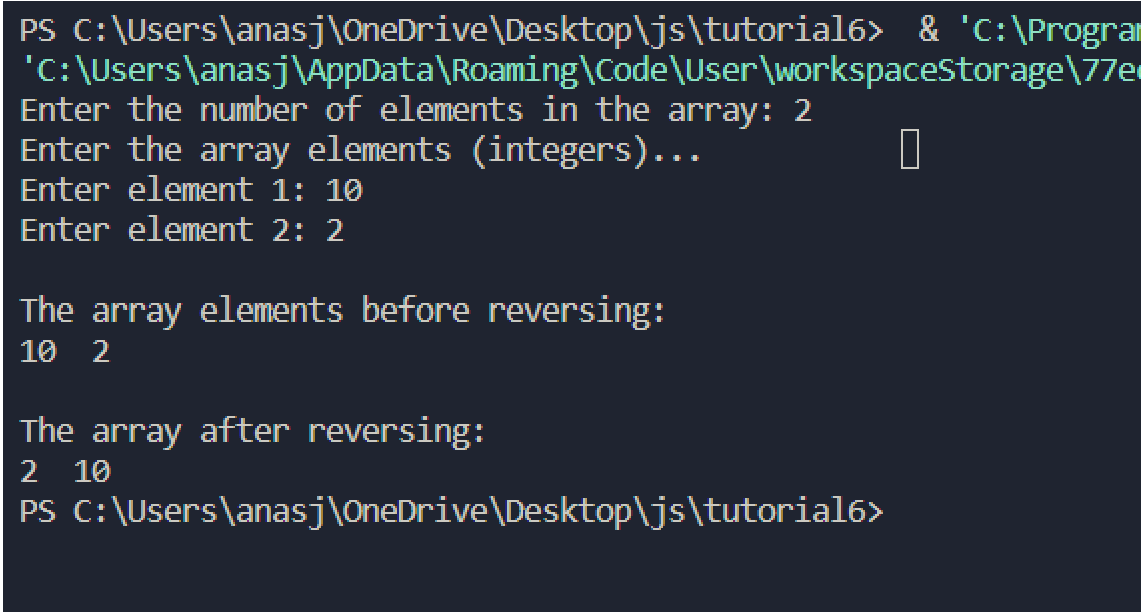
```
        while (true) {
        System.out.print("Grade another quiz? (y/n): ");
        String choice = scan.next();
        if (!choice.equalsIgnoreCase("y")) {
            break;
        }

        }

    }

}
```



```
How many questions are on the quiz?
5
Please enter the solution of question 1 :
3
Please enter the solution of question 2 :
4
Please enter the solution of question 3 :
3
Please enter the solution of question 4 :
2
Please enter the solution of question 5 :
1
Enter answer1 :
1
Enter answer2 :
3
Enter answer3 :
3
Enter answer4 :
3
Enter answer5 :
3
Questions answered correctly: 1
Percentage Grade: 20.0%
Grade another quiz? (y/n): n
PS C:\Users\anasj\OneDrive\Desktop\js\tutorial6>  c:; cd 'c:\
CodeDetailsInExceptionMessages' '-cp' 'C:\Users\anasj\AppData
```

Question 2:

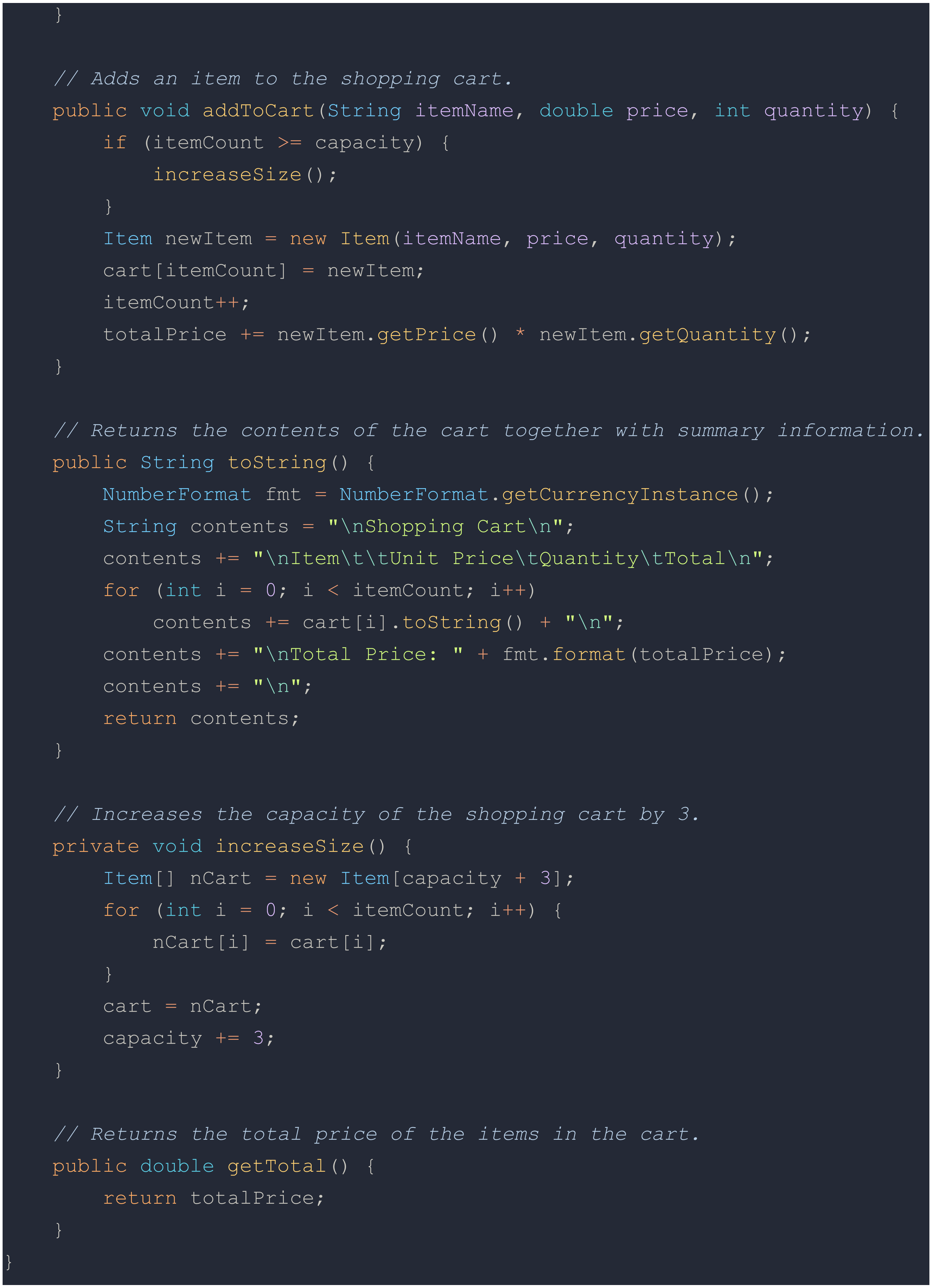
```
import java.util.Scanner;

public class q2{

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

    System.out.print ("Enter the number of elements in the array: ");
    numElements = scan.nextInt();

    int[] a = new int[numElements];

    System.out.println ("Enter the array elements (integers)...");
    for (int i = 0; i < numElements; i++)
    {
        System.out.print ("Enter element " + (i+1) + ": ");
        a[i] = scan.nextInt();
    }

    System.out.println ();
    System.out.println ("The array elements before reversing:");
    for (int i = 0; i < numElements; i++)
        System.out.print (a[i] + "  ");
    System.out.println();


    for (int i = 0; i < numElements/2; i++)
        {
        int temp = a[i];
        a[i] = a[numElements-1-i];
        a[numElements-1-i] = temp;
        }

    System.out.println ("\nThe array after reversing: ");
    for (int i = 0; i < numElements; i++)
        System.out.print (a[i] + "  ");
    System.out.println();
    scan.close();

        }

    }
```



```
PS C:\Users\anasj\OneDrive\Desktop\js\tutorial6>  & 'C:\Progra
'C:\Users\anasj\AppData\Roaming\Code\User\workspaceStorage\77e
Enter the number of elements in the array: 2
Enter the array elements (integers)...
Enter element 1: 10
Enter element 2: 2

The array elements before reversing:
10  2

The array after reversing:
2  10
PS C:\Users\anasj\OneDrive\Desktop\js\tutorial6>
```

Question 3:

```
import java.text.NumberFormat;

class ShoppingCart{
    private int itemCount; // total number of items in the cart
    private double totalPrice; // total price of items in the cart
    private int capacity; // current cart capacity
    private Item[] cart; // an array of Items

    // Creates an empty shopping cart with a capacity of 5 items.
    public ShoppingCart() {
        capacity = 5;
        itemCount = 0;
        totalPrice = 0.0;
        cart = new Item[capacity];
```



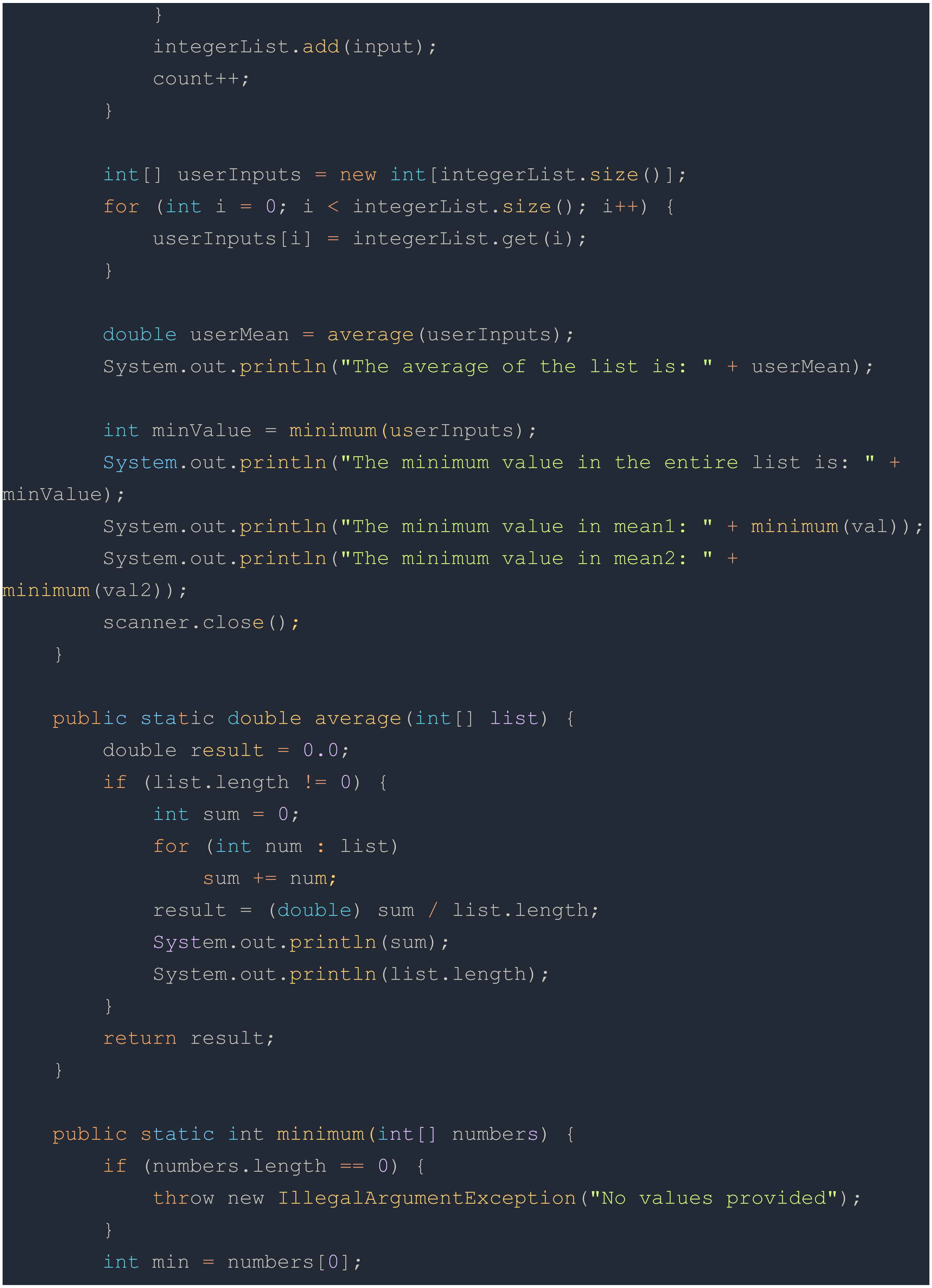
```java
    }

    // Adds an item to the shopping cart.
    public void addToCart(String itemName, double price, int quantity) {
        if (itemCount >= capacity) {
            increaseSize();
        }
        Item newItem = new Item(itemName, price, quantity);
        cart[itemCount] = newItem;
        itemCount++;
        totalPrice += newItem.getPrice() * newItem.getQuantity();
    }

    // Returns the contents of the cart together with summary information.
    public String toString() {
        NumberFormat fmt = NumberFormat.getCurrencyInstance();
        String contents = "\nShopping Cart\n";
        contents += "\nItem\t\tUnit Price\tQuantity\tTotal\n";
        for (int i = 0; i < itemCount; i++)
            contents += cart[i].toString() + "\n";
        contents += "\nTotal Price: " + fmt.format(totalPrice);
        contents += "\n";
        return contents;
    }

    // Increases the capacity of the shopping cart by 3.
    private void increaseSize() {
        Item[] nCart = new Item[capacity + 3];
        for (int i = 0; i < itemCount; i++) {
            nCart[i] = cart[i];
        }
        cart = nCart;
        capacity += 3;
    }

    // Returns the total price of the items in the cart.
    public double getTotal() {
        return totalPrice;
    }
}
```

Question 3:

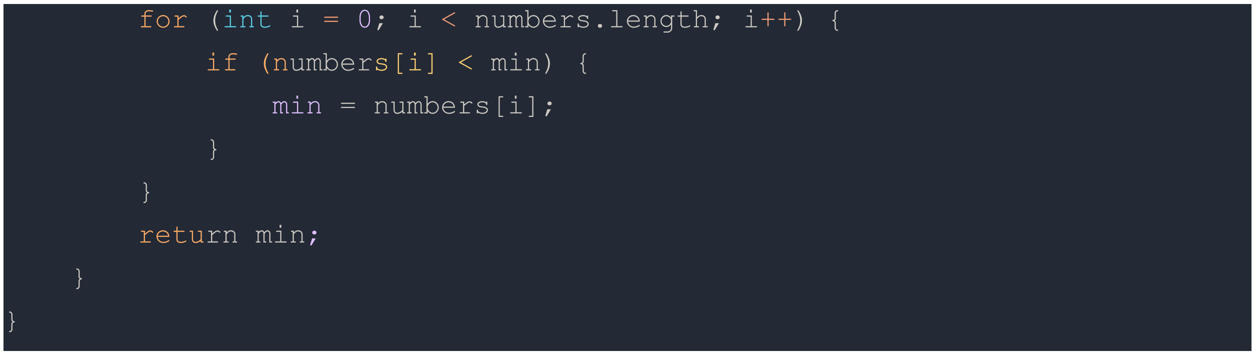
```
import java.text.NumberFormat;
import java.util.Scanner;

public class shoppingsimulation {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Item[] cart = new Item[20];             // Assuming a maximum of
20 items in the cart
        int itemCount = 0;

        while (true) {
            // Read item details from the user.
            System.out.print("Enter the name of the item: (Type 'quit' to
exit code)");
            String itemName = scanner.next();

            if (itemName.equals("quit")) {
                break; // Exit the loop if the user enters 'q'.
            }

            System.out.print("Enter the price of the item: ");
            double itemPrice = scanner.nextDouble();

            System.out.print("Enter the quantity of the item: ");
            int quantity = scanner.nextInt();

            // Add the item to the shopping cart.
            Item item = new Item(itemName, itemPrice, quantity);
            cart[itemCount] = item;
            itemCount++;

            // Print the contents of the cart.
            System.out.println("The Cart Contents are:");
            for (int i = 0; i < itemCount; i++) {
                System.out.println(cart[i].toString());
            }

            // Calculate and print the total price.
            double totalPrice = 0.0;
            for (int i = 0; i < itemCount; i++) {
                totalPrice += cart[i].getPrice() * cart[i].getQuantity();
            }

            NumberFormat fmt = NumberFormat.getCurrencyInstance();
            System.out.println("Please pay: " + fmt.format(totalPrice));
            }


        scanner.close();
    }
}
```

Question 4:

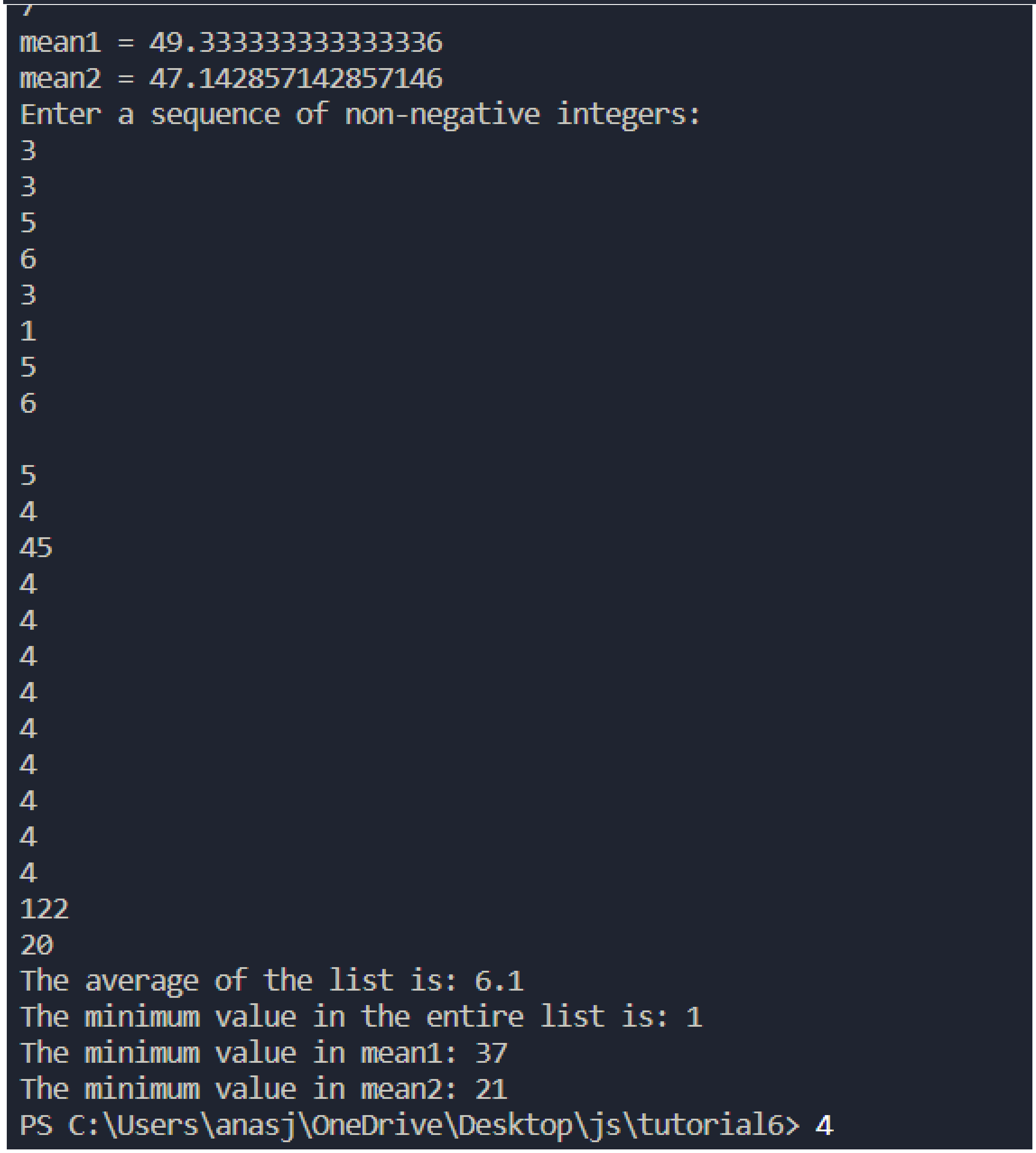
```
import java.util.ArrayList;
import java.util.Scanner;

class Parameters{
    public static void main(String[] args) {
        int[] val = {42, 69, 37};
        int[] val2 = {35, 43, 93, 23, 40, 21, 75};
        double mean1 = average(val);
        double mean2 = average(val2);
        System.out.println("mean1 = " + mean1);
        System.out.println("mean2 = " + mean2);

        Scanner scanner = new Scanner(System.in);
        ArrayList<Integer> integerList = new ArrayList<>();

        System.out.println("Enter a sequence of non-negative integers: ");

        int input;
        int count = 0;

        while (count < 20) {
            input = scanner.nextInt();
            if (input <= 0) {
                break;
```



```
            }
            integerList.add(input);
            count++;
        }

        int[] userInputs = new int[integerList.size()];
        for (int i = 0; i < integerList.size(); i++) {
            userInputs[i] = integerList.get(i);
        }

        double userMean = average(userInputs);
        System.out.println("The average of the list is: " + userMean);

        int minValue = minimum(userInputs);
        System.out.println("The minimum value in the entire list is: " +
minValue);
        System.out.println("The minimum value in mean1: " + minimum(val));
        System.out.println("The minimum value in mean2: " +
minimum(val2));
        scanner.close();
    }

    public static double average(int[] list) {
        double result = 0.0;
        if (list.length != 0) {
            int sum = 0;
            for (int num : list)
                sum += num;
            result = (double) sum / list.length;
            System.out.println(sum);
            System.out.println(list.length);
        }
        return result;
    }

    public static int minimum(int[] numbers) {
        if (numbers.length == 0) {
            throw new IllegalArgumentException("No values provided");
        }
        int min = numbers[0];
```



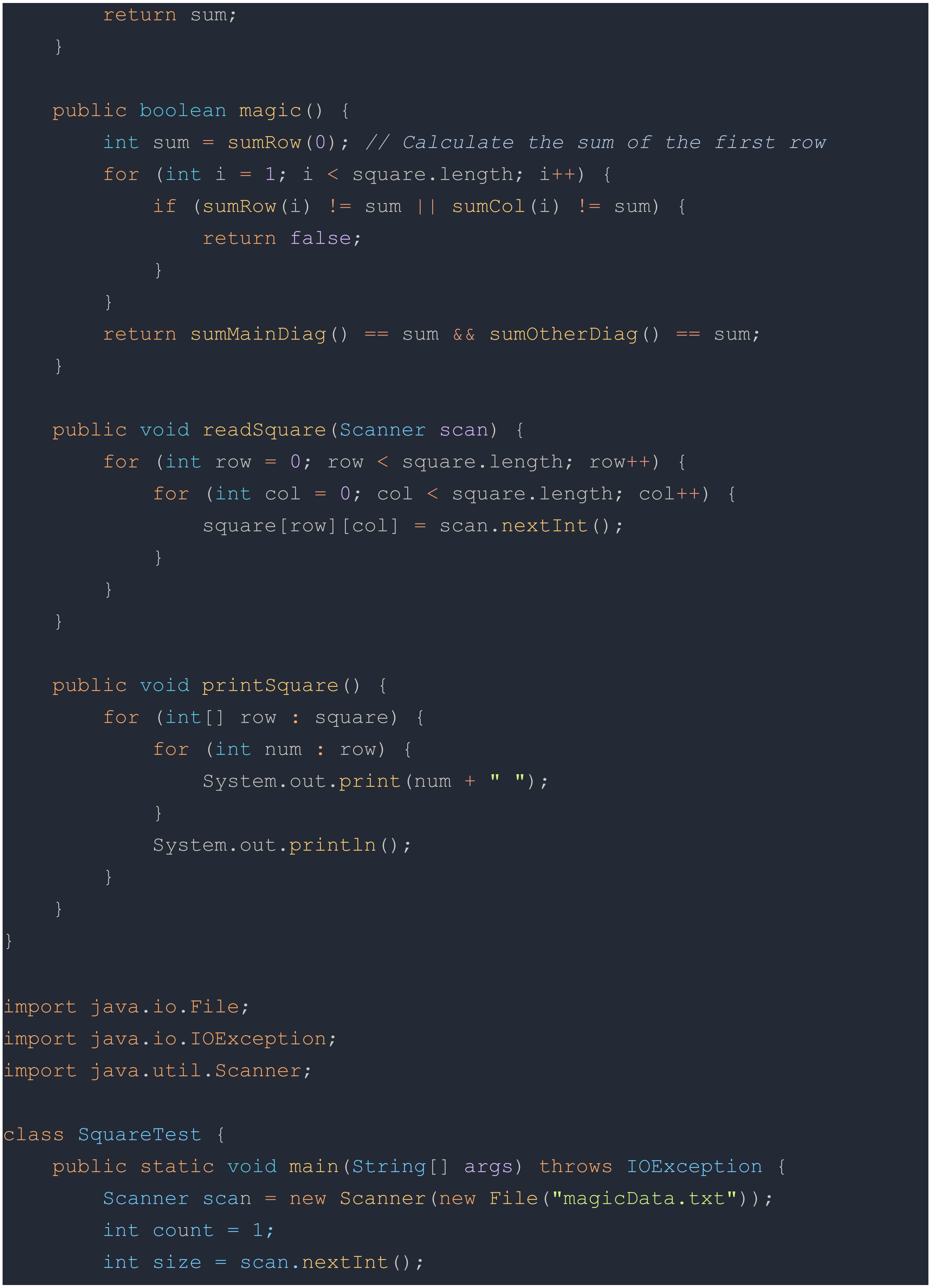
```
        for (int i = 0; i < numbers.length; i++) {
            if (numbers[i] < min) {
                min = numbers[i];
            }
        }
        return min;
    }
}
```



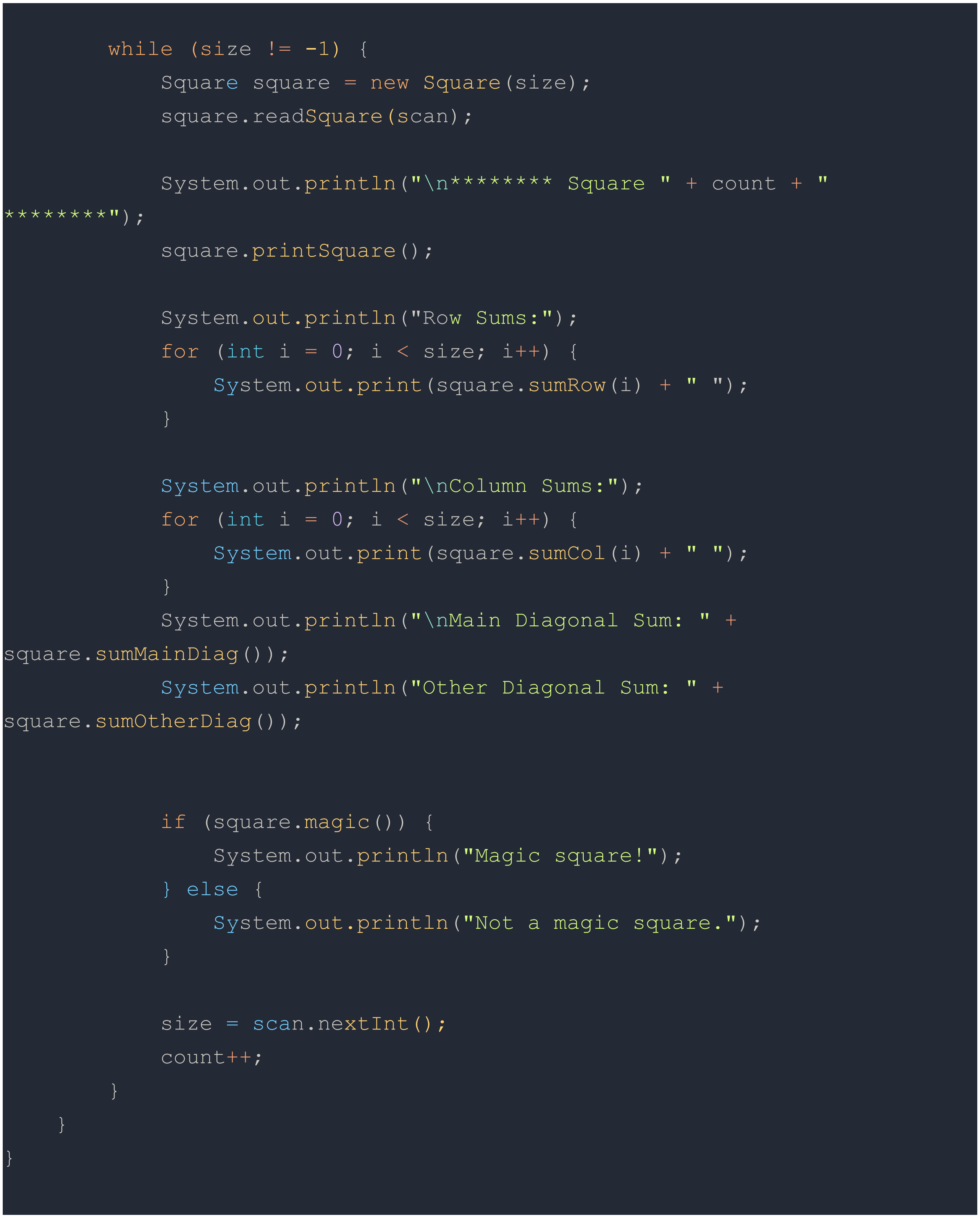
```
7
mean1 = 49.333333333333336
mean2 = 47.142857142857146
Enter a sequence of non-negative integers:
3
3
5
6
3
1
5
6

5
4
45
4
4
4
4
4
4
4
4
4
122
20
The average of the list is: 6.1
The minimum value in the entire list is: 1
The minimum value in mean1: 37
The minimum value in mean2: 21
PS C:\Users\anasj\OneDrive\Desktop\js\tutorial6> 4
```

Question 5:

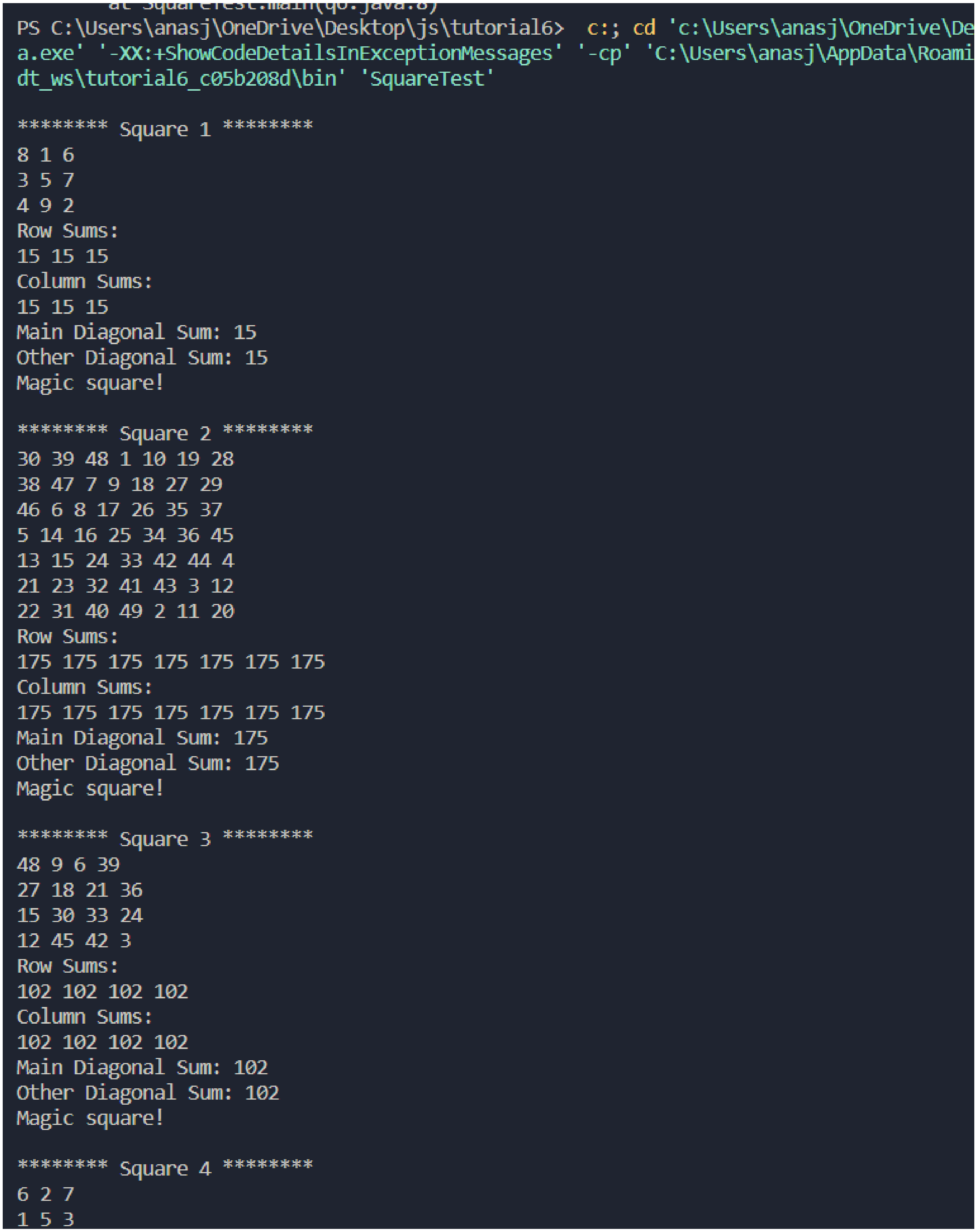
```
import java.util.Scanner;

class Square {
    private int[][] square;

    public Square(int size) {
        square = new int[size][size];
    }

    public int sumRow(int row) {
        int sum = 0;
        for (int col = 0; col < square.length; col++) {
            sum += square[row][col];
        }
        return sum;
    }

    public int sumCol(int col) {
        int sum = 0;
        for (int row = 0; row < square.length; row++) {
            sum += square[row][col];
        }
        return sum;
    }

    public int sumMainDiag() {
        int sum = 0;
        for (int i = 0; i < square.length; i++) {
            sum += square[i][i];
        }
        return sum;
    }

    public int sumOtherDiag() {
        int sum = 0;
        for (int i = 0; i < square.length; i++) {
            sum += square[i][square.length - 1 - i];
        }
```



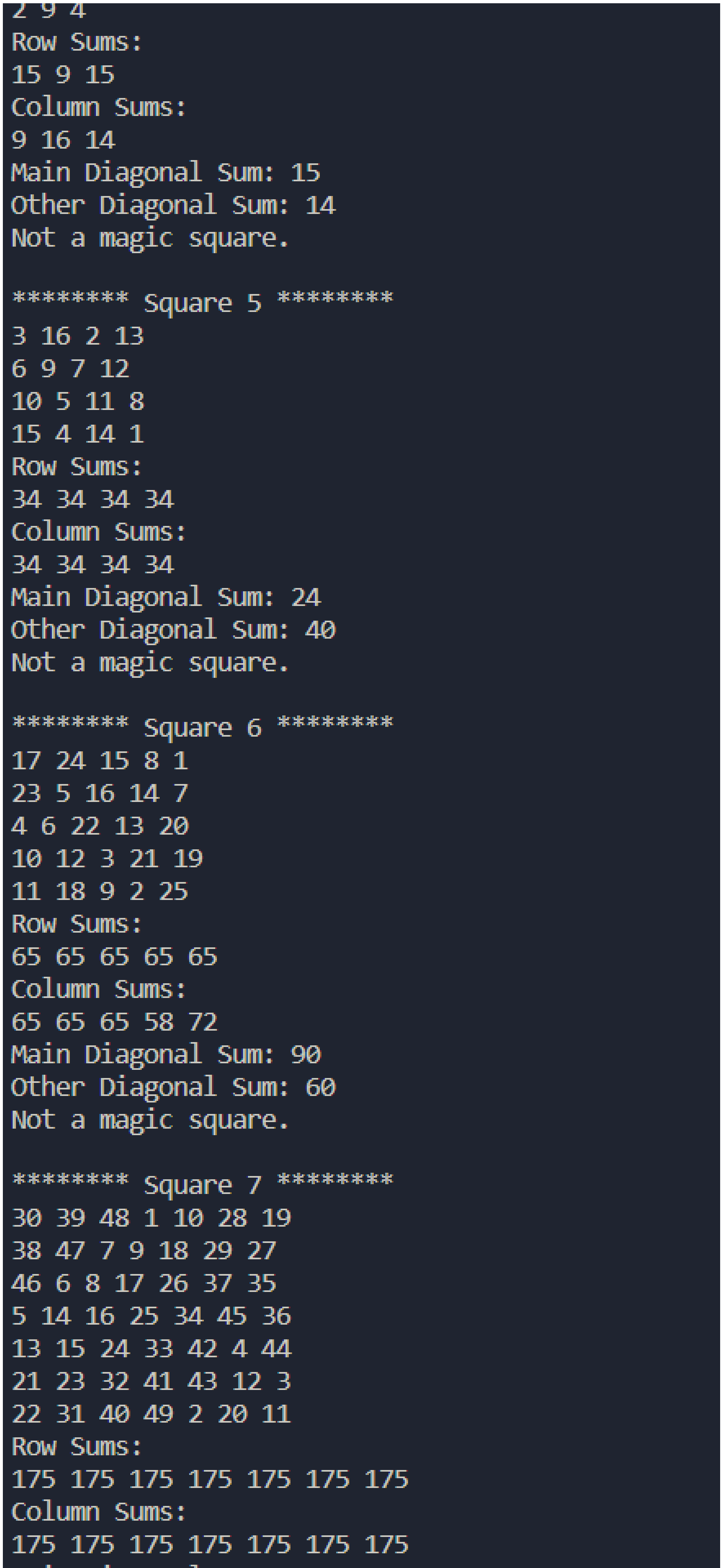
```java
        return sum;
    }

    public boolean magic() {
        int sum = sumRow(0); // Calculate the sum of the first row
        for (int i = 1; i < square.length; i++) {
            if (sumRow(i) != sum || sumCol(i) != sum) {
                return false;
            }
        }
        return sumMainDiag() == sum && sumOtherDiag() == sum;
    }

    public void readSquare(Scanner scan) {
        for (int row = 0; row < square.length; row++) {
            for (int col = 0; col < square.length; col++) {
                square[row][col] = scan.nextInt();
            }
        }
    }

    public void printSquare() {
        for (int[] row : square) {
            for (int num : row) {
                System.out.print(num + " ");
            }
            System.out.println();
        }
    }
}

import java.io.File;
import java.io.IOException;
import java.util.Scanner;

class SquareTest {
    public static void main(String[] args) throws IOException {
        Scanner scan = new Scanner(new File("magicData.txt"));
        int count = 1;
        int size = scan.nextInt();
```



```
        while (size != -1) {
            Square square = new Square(size);
            square.readSquare(scan);

            System.out.println("\n******** Square " + count + "
********");
            square.printSquare();

            System.out.println("Row Sums:");
            for (int i = 0; i < size; i++) {
                System.out.print(square.sumRow(i) + " ");
            }

            System.out.println("\nColumn Sums:");
            for (int i = 0; i < size; i++) {
                System.out.print(square.sumCol(i) + " ");
            }
            System.out.println("\nMain Diagonal Sum: " +
square.sumMainDiag());
            System.out.println("Other Diagonal Sum: " +
square.sumOtherDiag());


            if (square.magic()) {
                System.out.println("Magic square!");
            } else {
                System.out.println("Not a magic square.");
            }

            size = scan.nextInt();
            count++;
        }
    }
}
```



```
        at SquareTest.main(q6.java:8)
PS C:\Users\anasj\OneDrive\Desktop\js\tutorial6>  c:; cd 'c:\Users\anasj\OneDrive\De
a.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\anasj\AppData\Roami
dt_ws\tutorial6_c05b208d\bin' 'SquareTest'

******** Square 1 ********
8 1 6
3 5 7
4 9 2
Row Sums:
15 15 15
Column Sums:
15 15 15
Main Diagonal Sum: 15
Other Diagonal Sum: 15
Magic square!

******** Square 2 ********
30 39 48 1 10 19 28
38 47 7 9 18 27 29
46 6 8 17 26 35 37
5 14 16 25 34 36 45
13 15 24 33 42 44 4
21 23 32 41 43 3 12
22 31 40 49 2 11 20
Row Sums:
175 175 175 175 175 175 175
Column Sums:
175 175 175 175 175 175 175
Main Diagonal Sum: 175
Other Diagonal Sum: 175
Magic square!

******** Square 3 ********
48 9 6 39
27 18 21 36
15 30 33 24
12 45 42 3
Row Sums:
102 102 102 102
Column Sums:
102 102 102 102
Main Diagonal Sum: 102
Other Diagonal Sum: 102
Magic square!

******** Square 4 ********
6 2 7
1 5 3
```



```
2 9 4
Row Sums:
15 9 15
Column Sums:
9 16 14
Main Diagonal Sum: 15
Other Diagonal Sum: 14
Not a magic square.

******** Square 5 ********
3 16 2 13
6 9 7 12
10 5 11 8
15 4 14 1
Row Sums:
34 34 34 34
Column Sums:
34 34 34 34
Main Diagonal Sum: 24
Other Diagonal Sum: 40
Not a magic square.

******** Square 6 ********
17 24 15 8 1
23 5 16 14 7
4 6 22 13 20
10 12 3 21 19
11 18 9 2 25
Row Sums:
65 65 65 65 65
Column Sums:
65 65 65 58 72
Main Diagonal Sum: 90
Other Diagonal Sum: 60
Not a magic square.

******** Square 7 ********
30 39 48 1 10 28 19
38 47 7 9 18 29 27
46 6 8 17 26 37 35
5 14 16 25 34 45 36
13 15 24 33 42 4 44
21 23 32 41 43 12 3
22 31 40 49 2 20 11
Row Sums:
175 175 175 175 175 175 175
Column Sums:
175 175 175 175 175 175 175
```