12.3

Import java.io.\*;

Import java.util.\*;

Public class Test {

Public static void main(String[] args) {

Scanner input = new Scanner(System.in);

Int[] array = getArray();

System.out.print(“Enter the index of the array: “);

Try {

System.out.println(“The corresponding element value is “ +

Array[input.nextInt()]);

}

Catch (ArrayIndexOutOfBoundsException ex) {

System.out.println(“Out of Bounds.”);

}

}

Public static int[] getArray() {

Int[] array = new int[100];

For (int I = 0; I < array.length; i++) {

Array[i] = (int)(Math.random() \* 100) + 1;

}

Return array;

}

}

12.5

Import java.io.\*;

Import java.util.\*;

Public class test {

Public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print(“Enter a color: “);

String color = input.next();

System.out.print(“Is the triangle filled (true / false)? “);

Boolean filled = input.nextBoolean();

Boolean repeatSidesInput = true;

Do {

System.out.print(“Enter three side of the triangle: “);

Double side1 = input.nextDouble();

Double side2 = input.nextDouble();

Double side3 = input.nextDouble();

Try {

Triangle triangle = new Triangle(side1, side2, side3);

repeatSidesInput = false;

triangle.setColor(color);

triangle.setFilled(filled);

System.out.println(triangle.toString());

System.out.println(“Area: “ + triangle.getArea());

System.out.println(“Perimeter: “ + triangle.getPerimeter());

System.out.println(“Color: “ + triangle.getColor());

System.out.println(“Triangle is” + (triangle.isFilled() ? “” : “ not “)

+ “filled”);

}

Catch (IllegalTriangleException ex) {

System.out.println(ex.getMessage());

}

} while (repeatSidesInput);

}

}

Class GeometricObject {

Private String color = “white”;

Private boolean filled;

Private java.util.Date dateCreated;

Public GeometricObject() {

dateCreated = new java.util.Date();

}

Public GeometricObject(String color, boolean filled) {

dateCreated = new java.util.Date();

this.color = color;

this.filled = filled;

}

Public String getColor() {

Return color;

}

Public void setColor(String color) {

This.color = color;

}

Public boolean isFilled() {

Return filled;

}

Public void setFilled(boolean filled) {

This.filled = filled;

}

Public java.util.Date getDateCreated() {

Return dateCreated;

}

Public String toString() {

Return “created on “ + dateCreated + “\ncolor: “ + color +

“ and filled: “ + filled;

}

}

12.9

Import java.io.\*;

Import java.util.\*;

Public class test {

Public static void main(String[] args) {

System.out.println(bin2Dec(“1100100”));

System.out.println(bin2Dec(“lafkja”));

}

Public static int bin2Dec(String binary) throws BinaryFormatException {

If (!isBinary(binary)) {

Throw new BinaryFormatException(binary + “ is not a binary number.”);

}

Int power = 0;

Int decimal = 0;

For (int I = binary.length() – 1; I >= 0; i--) {

If (binary.charAt(i) == ‘1’) {

Decimal += Math.pow(2, power);

}

Power++;

}

Return decimal;

}

Public static boolean isBinary(String binary) {

For (char ch : binary.toCharArray()) {

If (ch != ‘1’ && ch != ‘0’) return false;

}

Return true;

}

}

Class BinaryFormatException extends IllegalArgumentException {

BinaryFormatException(String s) {

Super(s);

}

}

12.13

Import java.io.\*;

Import java.util.\*;

Public class test {

Public static void main(String[] args) throws FileNotFoundException {

If (args.length != 1) {

System.out.println(“Usage: java filename”);

System.exit(1);

}

File file = new File(args[0]);

If (!file.exists()) {

System.out.println(“File “ + args[0] + “ does not exist”);

System.exit(2);

}

Int characters = 0;

Int words = 0;

Int lines = 0;

Try (

Scanner input = new Scanner(file);

) {

While (input.hasNext()) {

Lines++;

String line = input.nextLine();

Characters += line.length();

}

}

Try (

Scanner input = new Scanner(file);

) {

While (input.hasNext()) {

String line = input.next();

Words++;

}

}

System.out.println(“File “ + file.getName() + “ has”);

System.out.println(characters + “ characters”);

System.out.println(words + “ words”);

System.out.println(lines + “ lines”);

}

}

12.15

Import java.io.\*;

Import java.util.\*;

Public class test {

Public static void main(String[] args) throws FileNotFoundException {

File file = new File(“Exercise12\_15.txt”);

If (file.exists()) {

System.out.println(“File already exists”);

System.exit(0);

}

Try (

PrintWriter output = new PrintWriter(file);

) {

For (int I = 0; I < 100; i++) {

Output.print(((int)(Math.random() \* 500) + 1));

Output.print(“ “);

}

}

ArrayList<Integer> list = new ArrayList<>();

Try(

Scanner input = new Scanner(file);

) {

While (input.hasNext()) {

List.add(input.nextInt());

}

}

Collections.sort(list);

System.out.print(list.toString());

System.out.println();

}

}

}

12.19

Import java.io.\*;

Import java.util.\*;

Public class test {

Public static void main(String[] args) throws FileNotFoundException {

Try {

// Create a URL object

Java.net.URL url = new java.net.URL(

<http://cs.armstrong.edu/liang/data/Lincoln.txt>);

int count = 0; // Counts words

// Open an input stream and create a Scanner object

Scanner input = new Scanner(url.openStream());

While (input.hasNext()) {

// Read words

If (Character.isLetter((input.next()).charAt(0))) {

Count++;

}

}

// Display result

System.out.println(

“Number of words in President Abraham Lincoln’s Gettysburg address: “ +

Count);

}

Catch (java.net.MalformedURLException ex) {

System.out.println(“Invalid URL”);

}

Catch (java.io.IOException ex) {

System.out.println(“I/0 Errors: no such file”);

}

}

}