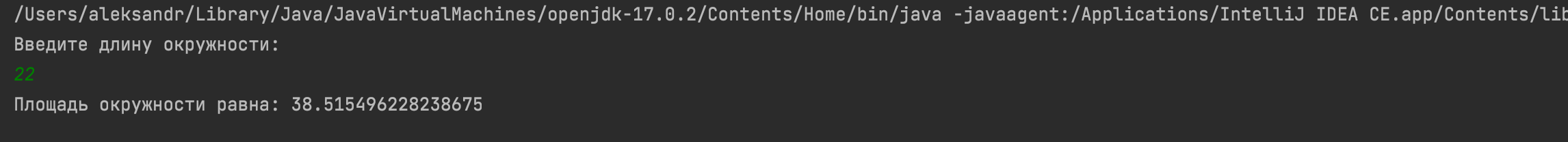
Контрольная работа Дурнев Александр

Код работы программы задания 1 и 2

package com.company;  
  
import java.util.Arrays;  
import java.util.Scanner;  
  
public class cyclicCase {  
 // Задание 2  
 //Реализовать процедуру (статический метод)  
 // циклического сдвига одномерного массива целых чисел вправо на n позиций.  
 //Пример работы:  
 //вход процедуры > arr = [1, 2, 3, 4, 5], n = 1  
 //выход > arr = [5, 1, 2, 3, 4]  
 //вход процедуры > arr = [1, 2, 3, 4, 5], n = 3  
 //выход > arr = [3, 4, 5, 1, 2]  
 public static void getArrayShift(int[] arr, int n) {  
 for (int i = 0; i < n; i++) {  
 int tmp = arr[arr.length - 1];  
 arr[arr.length - 1] = arr[0];  
 for (int j = 1; j < arr.length; j++) {  
 arr[arr.length - j] = arr[arr.length - j - 1];  
 }  
 arr[0] = tmp;  
 }  
 }  
  
  
 public static void main(String[] args) {  
 //Задание 1  
 //Пользователь вводит длину окружности (допускаются дробные значения).  
 // Программа должна вывести площадь заданной окружности.  
  
 //Решение:  
 //Создаем сканер для ввода данных  
 Scanner in = new Scanner(System.*in*);  
 {  
 double length; //объявляем переменную  
 System.*out*.println("Введите длину окружности: "); //запрашиваем пользователя ввести длину  
 length = in.nextDouble();  
 in.close(); //закрываем сканер  
 //Считаем площадь окружности  
 double square = (length \* length) / (4 \* Math.*PI*);  
 System.*out*.println("Площадь окружности равна: " + square);  
 }  
 System.*out*.println('\n'); //  
  
 //Решение второй задачи:  
 int arr[] = {1, 2, 3, 4, 5,};  
 System.*out*.println("Массив " + Arrays.*toString*(arr));  
 *getArrayShift*(arr, 1);  
 System.*out*.println("Со сдвигом на 1 " + Arrays.*toString*(arr));  
 *getArrayShift*(arr, 4);  
 System.*out*.println("Со сдвигом на 4 " + Arrays.*toString*(arr));  
 *getArrayShift*(arr, 3);  
 System.*out*.println("Со сдвигом на 3 " + Arrays.*toString*(arr));  
 }  
}

Тесты работы программы:



Изображение выглядит как текст

Автоматически созданное описание

Код программы задания 3

package com.company;  
  
import java.util.Scanner;  
  
public class Triangle {  
  
 //1. Создаем поля класса  
 private String name; //переменная имени  
 private double a; //переменная стороны А  
 private double b; //переменная стороны В  
 private double c; //переменная стороны С  
  
 // Создаем конструктор без параметров  
 public Triangle() {  
 this.name = "Name";  
 this.a = 1;  
 this.b = 1;  
 this.c = 1;  
 }  
  
 //2. Создаем конструктор с 4-мя параметрами  
 public Triangle(String name, double a, double b, double c) {  
 this.name = name;  
 this.a = a;  
 this.b = b;  
 this.c = c;  
 }  
  
 //3. Getters  
 public String getName() {  
 return name;  
 }  
  
 public double getA() {  
 return a;  
 }  
  
 public double getB() {  
 return b;  
 }  
  
 public double getC() {  
 return c;  
 }  
  
 //4. Setters  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 public void setA(double a) {  
 this.a = a;  
 }  
  
 public void setB(double b) {  
 this.b = b;  
 }  
  
 public void setC(double c) {  
 this.c = c;  
 }  
  
 //5. Вывод метода на экран  
 public void printTriangle() {  
 System.*out*.println(toString());  
 }  
  
 //6. Метод представления класса в виде строки - перегрузка метода toString  
 @Override  
 public String toString() {  
 return name + " сторона А = " + a + " сторона B = " + b + " сторона C = " + c;  
 }  
  
 //7. Метод ввода данных с клавиатуры  
 public void inputTriangle(Scanner in) {  
 System.*out*.print("Введите имя треугольника: ");  
 name = in.nextLine();  
 System.*out*.print("Введите сторону треугольника А: ");  
 a = in.nextDouble();  
 System.*out*.print("Введите сторону треугольника B: ");  
 b = in.nextDouble();  
 System.*out*.print("Введите сторону треугольника C: ");  
 c = in.nextDouble();  
 }  
  
 // 8. Метод вычисления периметра  
 public double Perimeter() {  
 return a + b + c;  
 }  
  
 //9. Метод вычисления площади  
 public double Square() {  
 double per = (a + b + c) / 2;  
 return Math.*sqrt*(per \* (per - a) \* (per - b) \* (per - c));  
 // return Math.sqrt(Square() / 2 \* (Square() / 2 - a) \* (Square() / 2 - b) \* (Square() / 2 - c));  
 }  
  
 // 10. Метод проверяющий корректность значений стороне треугольника  
 // (может ли такой треугольник существовать)  
 public void being() {  
 if ((a + b > c) && (a + c > b) && ((b + c > a) && a >= 0 && b >= 0 && c >= 0)) {  
 System.*out*.println("Данный треуголник существует!");  
 } else  
 System.*out*.println("Данный треуголник не может существовать!");  
 }  
}

Код программы в Main

package com.company;  
  
import java.util.Scanner;  
  
public class TriangleRun {  
  
 // ЗАДАЧА:  
// Реализовать класс «Треугольник».  
// Треугольник характеризуется тремя сторонами (допускаются дробные значения).  
// Необходимо реализовать методы получения периметра и площади треугольника,  
// метод, проверяющий корректность значений стороне треугольника  
// (может ли такой треугольник существовать), ввод/вывод треугольника, setters/getters, конструкторы.  
// Протестировать класс через создание объектов и вызовы методов в коде  
// (менюшки и CLI/GUI делать не надо, просто вызовы методов).  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 Triangle triangle = new Triangle();  
  
 //Вводим с клавиатуры параметры  
 triangle.inputTriangle(in);  
 //Выводим на экран  
 triangle.printTriangle();  
 System.*out*.println(); // пробел для лучшей читаемости  
 // Выводим площадь  
 System.*out*.println("Площадь равна = " + triangle.Square());  
 System.*out*.println(); // пробел для лучшей читаемости  
 // Выводим периметр  
 System.*out*.println("Периметр равен = " + triangle.Perimeter());  
 System.*out*.println(); // пробел для лучшей читаемости  
 //Проверяем существует такой треугольник  
 triangle.being(); // проверка на существование треугольника  
 System.*out*.println(); // пробел для лучшей читаемости  
 // Задаем значения треуголника через setters  
 triangle.setName("Треугольник по параметрам setters");  
 triangle.setA(15);  
 triangle.setB(20);  
 triangle.setC(30);  
 //Выводим на экран getters  
 System.*out*.println("Выводим на экран заданный через setters треуголник");  
 System.*out*.println("Имя треугольника: " + triangle.getName() +  
 " сторона A = " + triangle.getA() +  
 " сторона B = " + triangle.getB() +  
 " сторона C = " + triangle.getC());  
 triangle.printTriangle();  
 triangle.being();  
 in.close();  
 }  
}

Тесты программы:

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание