МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Российский государственный университет им. А.Н. Косыгина (Технологии. Дизайн. Искусство)»

Кафедра автоматизированных систем обработки информации и управления

Отчет по лабораторной работе № 8

по дисциплине «Объектно-ориентированное программирование»

Тема: Перегрузка операторов

Выполнил: Гоянов Р.Р., группа МВА-220

Проверил: Кузьмина Т.М.

Цель работы

Научиться перегружать операторы для работы с объектами созданных классов.

Задание

Вариант 6:

a) + (а+с) сложение радиусов, возвращаемое значение - число.

b) \* (а\*с) цилиндр имеющий больший объем, тип возвращаемого значения совпадает с типом операндов.

c) <, > цилиндр больше, если его объем больше, меньше, если объем меньше.

d) -- уменьшает на единицу каждое поле, возвращаемое значение имеет тип операнда.

Код класса Cylinder

class Cylinder

{

private double height;

private double radius;

public double Height

{

get => height;

set => height = value;

}

public double Radius

{

get => radius;

set => radius = value;

}

public Cylinder(double height, double radius)

{

this.height = height;

this.radius = radius;

}

public double Volume => Math.PI \* radius \* radius \* height;

public bool Valid => radius >= 0 && height >= 0;

public override string ToString()

{

if (!Valid) return "Неверные данные";

return "Высота: " + height + "\nРадиус основания: " + radius + "\nОбъём: " + Math.Round(Volume, 3);

}

public static double operator+ (Cylinder firstCylinder, Cylinder secondCylinder)

{

return firstCylinder.radius + secondCylinder.radius;

}

public static Cylinder operator\* (Cylinder firstCylinder, Cylinder secondCylinder)

{

return firstCylinder.Volume > secondCylinder.Volume ? firstCylinder : secondCylinder;

}

public static bool operator> (Cylinder firstCylinder, Cylinder secondCylinder)

{

return firstCylinder.Volume > secondCylinder.Volume;

}

public static bool operator< (Cylinder firstCylinder, Cylinder secondCylinder)

{

return firstCylinder.Volume < secondCylinder.Volume;

}

public static Cylinder operator-- (Cylinder cylinder)

{

cylinder.radius--;

cylinder.height--;

return cylinder;

}

}

Код главного класса

public partial class MainWindow : Window

{

private Cylinder cyl1 = new Cylinder(0, 0);

private Cylinder cyl2 = new Cylinder(0,0);

public MainWindow()

{

InitializeComponent();

firstInfoArea.Text = cyl1.ToString();

secondInfoArea.Text = cyl2.ToString();

}

private double validateData(TextBox textBox)

{

double value = double.Parse(textBox.Text);

if (value < 0) throw new Exception();

return value;

}

private void firstHeightBox\_TextChanged(object sender, TextChangedEventArgs e)

{

try

{

cyl1.Height = validateData(firstHeightBox);

cyl1.Radius = validateData(firstRadiusBox);

firstInfoArea.Text = cyl1.ToString();

}

catch (Exception ex)

{

firstInfoArea.Text = "Неверные данные";

}

}

private void firstRadiusBox\_TextChanged(object sender, TextChangedEventArgs e)

{

try

{

cyl1.Height = validateData(firstHeightBox);

cyl1.Radius = validateData(firstRadiusBox);

firstInfoArea.Text = cyl1.ToString();

}

catch (Exception ex)

{

firstInfoArea.Text = "Неверные данные";

}

}

private void firstIncrementButton\_Click(object sender, RoutedEventArgs e)

{

cyl1--;

firstInfoArea.Text = cyl1.ToString();

}

private void secondHeightBox\_TextChanged(object sender, TextChangedEventArgs e)

{

try

{

cyl2.Height = validateData(secondHeightBox);

cyl2.Radius = validateData(secondRadiusBox);

secondInfoArea.Text = cyl2.ToString();

}

catch (Exception ex)

{

secondInfoArea.Text = "Неверные данные";

}

}

private void secondRadiusBox\_TextChanged(object sender, TextChangedEventArgs e)

{

try

{

cyl2.Height = validateData(secondHeightBox);

cyl2.Radius = validateData(secondRadiusBox);

secondInfoArea.Text = cyl2.ToString();

}

catch (Exception ex)

{

secondInfoArea.Text = "Неверные данные";

}

}

private void secondIncrementButton\_Click(object sender, RoutedEventArgs e)

{

cyl2--;

secondInfoArea.Text = cyl2.ToString();

}

private void plusButton\_Click(object sender, RoutedEventArgs e)

{

thirdInfoArea.Text = "Сумма радиусов равна " + (cyl1 + cyl2);

}

private void multiplyButton\_Click(object sender, RoutedEventArgs e)

{

thirdInfoArea.Text = "Информация о цилиндре с наибольшим объёмом:\n" + cyl1 \* cyl2;

}

private void compareButton\_Click(object sender, RoutedEventArgs e)

{

if (cyl1.Volume == cyl2.Volume) thirdInfoArea.Text = "Объёмы цилиндров равны.";

else thirdInfoArea.Text = "Объём первого цилиндра " + (cyl1 > cyl2 ? "больше" : "меньше") + " объёма второго.";

}

}

Результаты выполнения программы

Начальное состояние программы представлено на рис. 1

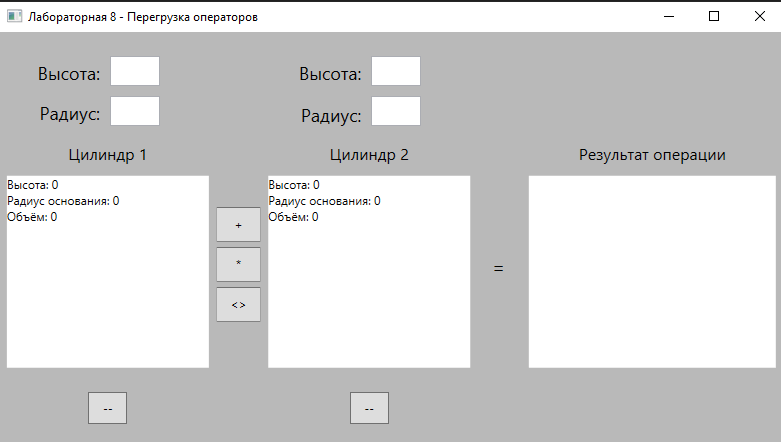


рис. 1

Пример состояния программы при вводе корректных и некорректных данных представлен на рис. 2

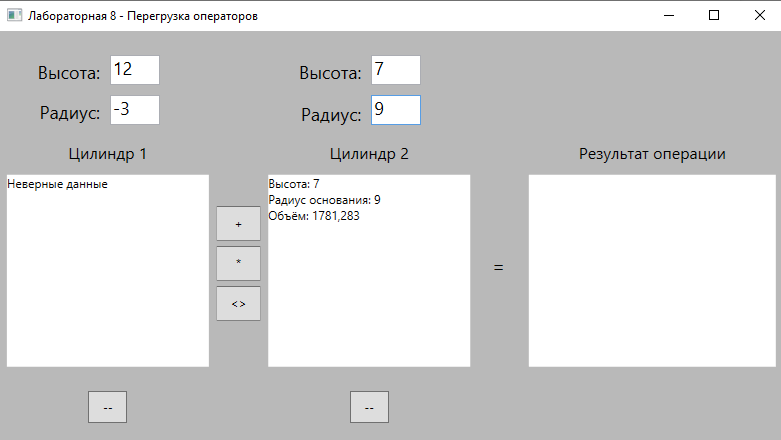


рис. 2

Состояние программы после нажатия на кнопку + представлено на рис. 3

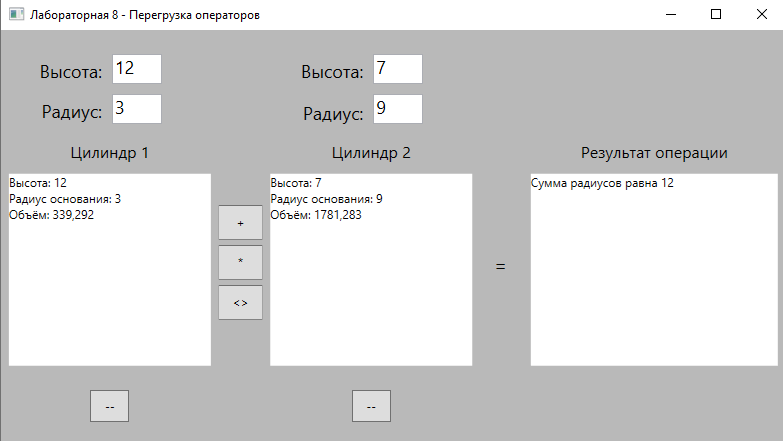


рис. 3

Состояние программы после нажатия на кнопку \* представлено на рис. 4

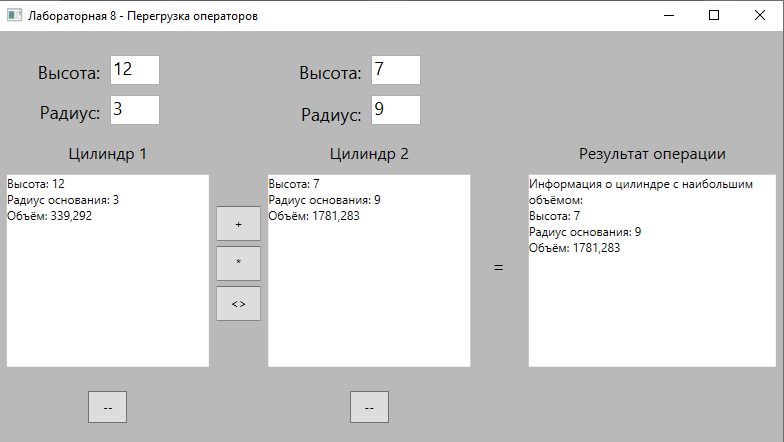


рис. 4

Состояние программы после нажатия на кнопку <> представлено на рис.5

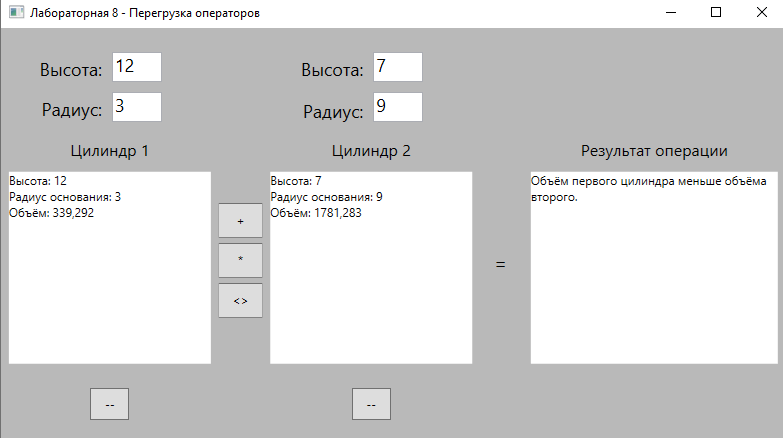


рис. 5

Состояние программы после нажатия на кнопку - - для первого цилиндра представлено на рис. 6

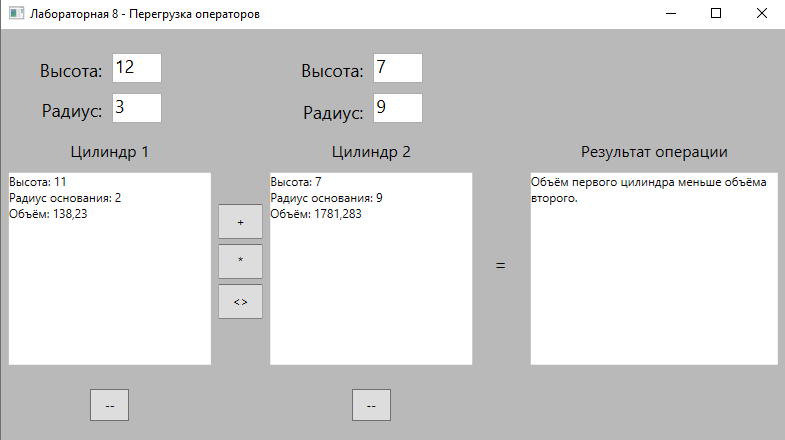


рис. 6