МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

ИНСТИТУТ НЕПРЕРЫВНОГО И ДИСТАНЦИОННОГО ОБРАЗОВАНИЯ

|  |
| --- |
| КАФЕДРА компьютерных технологий и программной инженерии |

ОЦЕНКА

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| КОНТРОЛЬНАЯ РАБОТА |
|  |
| по дисциплине: Компьютерная графика |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | Z9431 |  |  |  | Андреев Д.И. |
|  | номер группы |  | подпись, дата |  | инициалы, фамилия |
| Студенческий билет № | 2019/3781 | |  |  |  |

|  |  |
| --- | --- |
| Шифр ИНДО |  |

Санкт-Петербург 2020

Оглавление

[1. Вариант задания 3](#_Toc55750644)

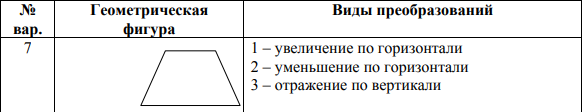
[2. Листинг программы 4](#_Toc55750645)

[3. Объяснение результатов и выводы по работы 5](#_Toc55750646)

1. Вариант задания

Используя библиотеку GDI+ реализовать построение геометрической фигуры и аффинные преобразования на плоскости.

Вариант 7



1. Листинг программы

Класс формы

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

this.label1.Text = "Контрольная работа.\nВариант " + ((81 % 15) + 1).ToString();

// Координаты трапеции задаются относительно ноля

Trapeze = new PointF[4]

{

new PointF( -50, 50),

new PointF( 100, 50),

new PointF( 200, -50),

new PointF(-100, -50),

};

}

protected void OnPaint(object sender, PaintEventArgs e)

{

// Получаем объект Graphics через идентификатор окна

Graphics g = Graphics.FromHwnd(this.Handle);

// А также ширину и высоту области для рисования

int w = ClientSize.Width;

int h = ClientSize.Height;

// Очистка формы с помощью закрашенного прямоугольника

g.FillRectangle(Brushes.White, 0, 0, w, h);

// Многоугольник, координаты трапеции преобразуются относительно экранных

PointF[] points = new PointF[4]

{

new PointF(Trapeze[0].X + w/2, Trapeze[0].Y + h/2),

new PointF(Trapeze[1].X + w/2, Trapeze[1].Y + h/2),

new PointF(Trapeze[2].X + w/2, Trapeze[2].Y + h/2),

new PointF(Trapeze[3].X + w/2, Trapeze[3].Y + h/2)

};

g.DrawPolygon(Pens.Green, points);

}

private void OnKeyPress(object sender, KeyPressEventArgs e)

{

//Получаем значение клавиша, которая была нажата

switch (e.KeyChar)

{

case 'w':

Trapeze[0].Y \*= 1.1f;

Trapeze[1].Y \*= 1.1f;

Trapeze[2].Y \*= 1.1f;

Trapeze[3].Y \*= 1.1f;

break;

case 's':

Trapeze[0].Y \*= 0.9f;

Trapeze[1].Y \*= 0.9f;

Trapeze[2].Y \*= 0.9f;

Trapeze[3].Y \*= 0.9f;

break;

case 'a':

case 'd':

Trapeze[0].X \*= -1;

Trapeze[1].X \*= -1;

Trapeze[2].X \*= -1;

Trapeze[3].X \*= -1;

break;

default:

break;

}

Invalidate();

}

private PointF[] Trapeze;

}

Инициализация формы

private void InitializeComponent()

{

this.label1 = new System.Windows.Forms.Label();

this.SuspendLayout();

//

// label1

//

this.label1.AutoSize = true;

this.label1.Location = new System.Drawing.Point(12, 20);

this.label1.Name = "label1";

this.label1.Size = new System.Drawing.Size(0, 13);

this.label1.TabIndex = 0;

//

// Form1

//

this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);

this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;

this.ClientSize = new System.Drawing.Size(800, 450);

this.Controls.Add(this.label1);

this.Name = "Form1";

this.Text = "Form1";

this.ResumeLayout(false);

this.KeyPress += this.OnKeyPress;

this.Paint += this.OnPaint;

this.PerformLayout();

}

1. Объяснение результатов и выводы по работы

Для решения поставленной задачи был использован массив точек типа PointF. При этом все преобразования координат происходят относительно абсолютного ноля. При отрисовке трапеции происходит переход от абсолютных координат к экранным, таким образом трапеция располагается в центре.

Преобразования происходят по нажатиям клавиш. При этом увеличение по вертикале дает умножение текущего X всех координат на коэффициент сжатия или растяжения, в зависимости от того, уменьшается размер фигуры или увеличивается. Отражение относительно горизонтальной оси происходит умножением координат всех точек на -1.