

Практическая работа 18 Методы работы с графикой

Цель занятия

Получить практические навыки разработки модулей для создания графических изображений.

Перечень оборудования и программного обеспечения

Персональный компьютер
Microsoft Office (Word, Visio)
Microsoft Visual Studio 2010

Краткие теоретические сведения

1 Рисование линий, кривых и фигур

Графические примитивы GDI+

Класс	Описание
Bitmap	Представляет растровые изображения
Font	Определяет атрибуты шрифта: размер, гарнитуру и т. д.
HatchBrush	Определяет заливку решетчатым узором
LinearGradientBrush	Определяет заливку с линейным градиентом
Pen	Определяет внешний вид линий и кривых
SolidBrush	Определяет заливку сплошным цветом
TextureBrush	Определяет заливку с использованием растрового изображения

Некоторые из этих классов определены в пространстве имен System.Drawing, другие — в пространстве имен System.Drawing.Drawing2D.

Перья, представленные экземплярами класса Pen, управляют внешним видом линий и кривых.

Объект Font задают внешний вид текста, тогда как кисти, представленные объектами классов HatchBrush, LinearGradientBrush, SolidBrush и TextureBrush определяют виды заливки. Приведенный ниже метод рисует прямоугольники трех стилей: 1) без заливки, 2) покрашенный красным цветом и 3) с градиентной заливкой, переходящей из красного в синее:

Пример:

```

Graphics im = CreateGraphics();
    // создание графической среды
Pen pen = new Pen(Color.Black);

// Нарисовать пустой прямоугольник.
im.DrawRectangle(pen, 10, 10, 390, 90);

// Нарисовать прямоугольник со сплошной заливкой.
SolidBrush solid = new SolidBrush (Color.Red);
im.FillRectangle (solid, 10, 110, 390, 90);

// Нарисовать прямоугольник с текстурной заливкой.

Image myImage = Image.FromFile("MyTexture.bmp");
TextureBrush myTextureBrush = new TextureBrush(myImage);
myGraphics.FillEllipse(myTextureBrush, 0, 0, 100, 50);

using System.Drawing.Drawing2D;
// Нарисовать прямоугольник с градиентной заливкой.
Rectangle rect = new Rectangle (10, 210, 390, 90);
LinearGradientBrush gradient = new LinearGradientBrush (rect, Color.Red,
Color.Blue, LinearGradientMode.Horizontal);
im.FillRectangle (gradient, rect);
im.DrawRectangle (pen, rect);

```

Объект Pen, передаваемый DrawRectangle, обводит каждый прямоугольник черной линией толщиной в 1 единицу. (По умолчанию 1 единица — это один пиксел, поэтому все линии в данном примере имеют ширину в 1 пиксел.) Кисти, передаваемые FillRectangle, управляют заливкой внутренней части прямоугольников. Заметьте: рамки и заливки рисуются по отдельности.

Класс Graphics содержит разнообразные открытые методы Draw и Fill для рисования линий, кривых, прямоугольников, эллипсов и других фигур.

Методы класса Graphics для рисования линий, кривых и фигур

Метод	Описание
DrawArc	Рисует дугу
DrawBezier	Рисует кривую Безье
DrawCurve	Рисует фундаментальный сплайн
DrawEllipse	Рисует круг или эллипс
DrawIcon	Рисует значок
DrawImage	Рисует картинку
DrawLine	Рисует линию
DrawPie	Рисует сектор

DrawPolygon	Рисует многоугольник
DrawString	Рисует строку текста
FillEllipse	Рисует круг или эллипс с заливкой
FillPie	Рисует сектор с заливкой
FillPolygon	Рисует многоугольник с заливкой
FillRectangle	Рисует прямоугольник с заливкой

2 Координаты и преобразования

При вызове DrawRectangle и FillRectangle вы определяете координаты, указывающие положение левого верхнего угла прямоугольника, и расстояния, задающие его ширину и высоту. По умолчанию расстояния измеряются в пикселах. Координаты задают расположение в двумерной системе, начало которой располагается в левом верхнем углу формы, а оси x и y направлены вправо и вниз соответственно.

Метод рисует прямоугольник шириной 200 и высотой 100 единиц в левом верхнем углу:

```
{
SolidBrush brush = new SolidBrush (Color.Red);
im.FillRectangle (brush, 0, 0, 200, 100);

//Вставка рисунка
Image newImage = Image.FromFile("Sample.jpg");
im.DrawImage(newImage, x, y);
}
SolidBrush blueBrush = new SolidBrush(Color.Blue);
// Create points that define polygon.
Point point1 = new Point(50, 50);
Point point2 = new Point(100, 25);
Point point3 = new Point(200, 5);
Point point4 = new Point(250, 50);
Point point5 = new Point(300, 100);
Point point6 = new Point(350, 200);
Point point7 = new Point(250, 250);
Point[] curvePoints = {point1, point2, point3, point4, point5, point6,
point7};

// Draw polygon to screen.
im.FillPolygon(blueBrush, curvePoints);

Rectangle rect = new Rectangle(10, 210, 90, 90);
LinearGradientBrush gradient = new LinearGradientBrush(rect, Color.Red,
Color.DarkOrange, LinearGradientMode.Horizontal);
im.FillEllipse(gradient, rect);
```

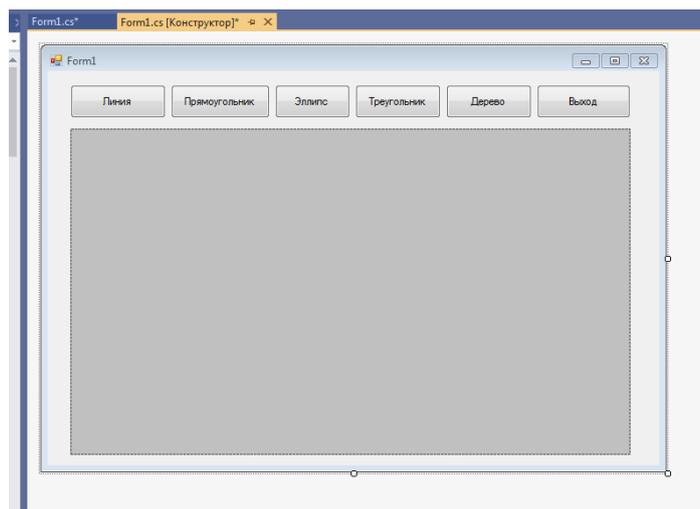
Задания

- 1 Изучить теоретические сведения и задание к работе
- 2 В соответствии с вариантом задания разработать и отладить программный модуль

Порядок выполнения работы (Пример выполнения)

Задание 2

- 1 Интерфейс формы в VisualStudio



- 2 Текст программы.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp5
{
    public partial class Form1 : Form
    {
        public Form1()
        {
```

```

InitializeComponent();
}

private void Form1_Load(object sender, EventArgs e)
{
}

private void pictureBox1_Click(object sender, EventArgs e)
{
}

private void button5_Click(object sender, EventArgs e)
{
Close();
}

private void button1_Click(object sender, EventArgs e)
{
Bitmap bbb = new Bitmap(pictureBox1.Width, pictureBox1.Height);
Graphics g = Graphics.FromImage(bbb);
Pen ppp = new Pen(Color.Red, 3);
g.DrawLine(ppp, 10, 50, 500, 150);
pictureBox1.Image = bbb;
}

private void button2_Click(object sender, EventArgs e)
{
Bitmap bbb = new Bitmap(pictureBox1.Width, pictureBox1.Height);
Graphics g = Graphics.FromImage(bbb);
Pen ppp = new Pen(Color.Blue, 4);
g.DrawRectangle(ppp, 100, 100, 300, 200);
pictureBox1.Image = bbb;
}

private void button3_Click(object sender, EventArgs e)
{
Bitmap bbb = new Bitmap(pictureBox1.Width, pictureBox1.Height);
Graphics g = Graphics.FromImage(bbb);
Pen ppp = new Pen(Color.SaddleBrown, 5);
g.DrawEllipse(ppp, 200, 100, 300, 200);
pictureBox1.Image = bbb;
}

private void button4_Click(object sender, EventArgs e)
{
Bitmap bbb = new Bitmap(pictureBox1.Width, pictureBox1.Height);
Graphics g = Graphics.FromImage(bbb);

```

```

Pen ppp = new Pen(Color.BlueViolet, 5);
Point point1 = new Point(150, 150);
Point point2 = new Point(200, 125);
Point point3 = new Point(300, 50);
Point[] curvePoints = {point1, point2, point3};
g.DrawPolygon(ppp, curvePoints);
pictureBox1.Image = bbb;
}
    private void button6_Click(object sender, EventArgs e)
{
    Bitmap bbb = new Bitmap(pictureBox1.Width, pictureBox1.Height);
    Graphics g = Graphics.FromImage(bbb);
    //СТВОЛ ДЕРЕВА
    SolidBrush stvol = new SolidBrush(Color.SaddleBrown);
    SolidBrush teny = new SolidBrush(Color.DimGray);
    g.FillEllipse(stvol, 300, 150, 100, 170);
    SolidBrush br = new SolidBrush(Color.Silver);
    g.FillEllipse(br, 365, 150, 100, 170);
    g.FillEllipse(teny, 255, 150, 100, 165);
    g.FillEllipse(br, 245, 150, 100, 170);
    g.FillEllipse(br, 330, 130, 40, 60);
    g.FillEllipse(br, 320, 300, 30, 40);
    g.FillEllipse(br, 352, 297, 30, 35);
    //Рисование кроны
    SolidBrush krd = new SolidBrush(Color.DarkGreen);
    SolidBrush kr = new SolidBrush(Color.ForestGreen);
    SolidBrush krl = new SolidBrush(Color.LawnGreen);
    g.FillEllipse(krd, 260, 140, 70, 70);
    g.FillEllipse(kr, 265, 135, 70, 70);
    g.FillEllipse(krl, 330, 45, 50, 50);
    g.FillEllipse(kr, 325, 50, 50, 50);
    g.FillEllipse(krd, 380, 160, 50, 50);
    g.FillEllipse(krl, 390, 150, 50, 50);
    g.FillEllipse(kr, 385, 155, 50, 50);
    g.FillEllipse(krd, 265, 120, 60, 60);
    g.FillEllipse(kr, 270, 125, 60, 60);
    g.FillEllipse(krl, 350, 70, 70, 70);
    g.FillEllipse(kr, 345, 75, 70, 70);
    g.FillEllipse(krl, 375, 100, 60, 60);
    g.FillEllipse(krd, 365, 115, 60, 60);
    g.FillEllipse(kr, 370, 105, 60, 60);
    g.FillEllipse(krd, 352, 135, 50, 50);
    g.FillEllipse(krl, 362, 125, 50, 50);
    g.FillEllipse(kr, 357, 130, 50, 50);
    g.FillEllipse(krd, 290, 125, 60, 60);

```

```

g.FillEllipse(krd, 285, 85, 60, 60);
g.FillEllipse(kr, 290, 80, 50, 50);
g.FillEllipse(krl, 325, 85, 60, 60);
g.FillEllipse(kr, 320, 90, 60, 60);
g.FillEllipse(kr, 357, 130, 50, 50);
g.FillEllipse(krl, 290, 115, 60, 60);
g.FillEllipse(krd, 280, 125, 60, 60);
g.FillEllipse(kr, 285, 120, 60, 60);
pictureBox1.Image = bbb;
}
}
}

```

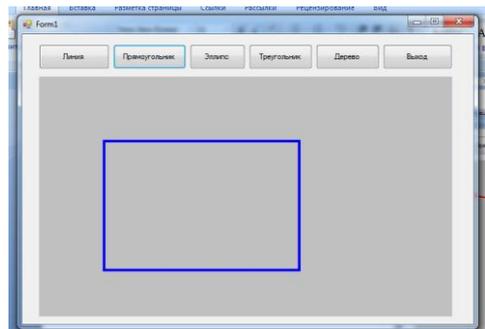
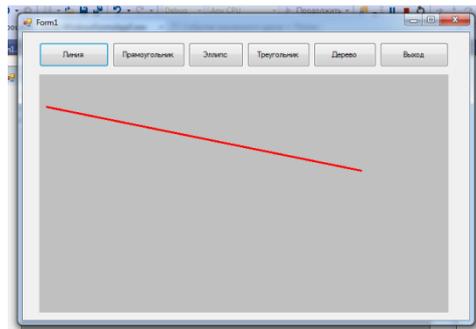
3 VisualStudio (фрагмент).

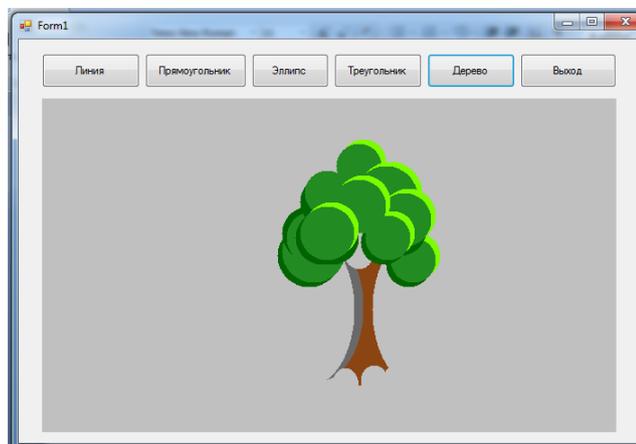
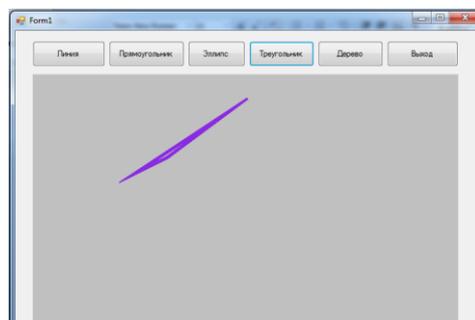
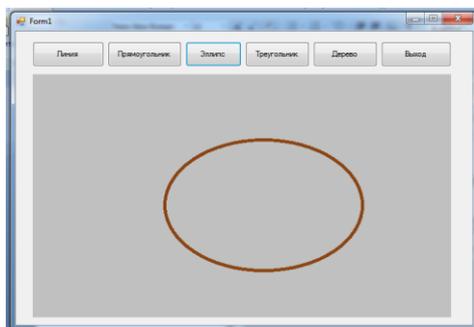
```

Form1.cs [Конструктор]
FormsApp5 - WindowsFormsApp5.Form1
Point point1 = new Point(150, 150);
Point point2 = new Point(200, 125);
Point point3 = new Point(300, 50);
Point[] curvePoints = {point1, point2, point3};
g.DrawPolygon(ppp, curvePoints);
pictureBox1.Image = bbb;
}
// ссылка: 1
private void button6_Click(object sender, EventArgs e)
{
    Bitmap bbb = new Bitmap(pictureBox1.Width, pictureBox1.Height);
    Graphics g = Graphics.FromImage(bbb);
    //Стол дерева
    SolidBrush stvol = new SolidBrush(Color.SaddleBrown);
    SolidBrush teny = new SolidBrush(Color.DimGray);
    g.FillEllipse(stvol, 300, 150, 100, 170);
    SolidBrush br = new SolidBrush(Color.Silver);
    g.FillEllipse(br, 365, 150, 100, 170);
    g.FillEllipse(teny, 255, 150, 100, 165);
    g.FillEllipse(br, 245, 150, 100, 170);
    g.FillEllipse(br, 330, 130, 40, 60);
    g.FillEllipse(br, 320, 300, 30, 40);
    g.FillEllipse(br, 352, 297, 30, 35);
    //Рисование кроны
    SolidBrush krd = new SolidBrush(Color.DarkGreen);
    SolidBrush kr = new SolidBrush(Color.ForestGreen);
    SolidBrush krl = new SolidBrush(Color.LawnGreen);
    g.FillEllipse(krd, 260, 140, 70, 70);
    g.FillEllipse(kr, 265, 135, 70, 70);
    g.FillEllipse(krl, 330, 45, 50, 50);
    g.FillEllipse(kr, 325, 50, 50, 50);
    g.FillEllipse(krd, 380, 160, 50, 50);
    g.FillEllipse(krl, 390, 150, 50, 50);
    g.FillEllipse(kr, 385, 155, 50, 50);
    g.FillEllipse(krd, 265, 120, 60, 60);
    g.FillEllipse(kr, 270, 125, 60, 60);
}

```

4 Результаты работы программы:



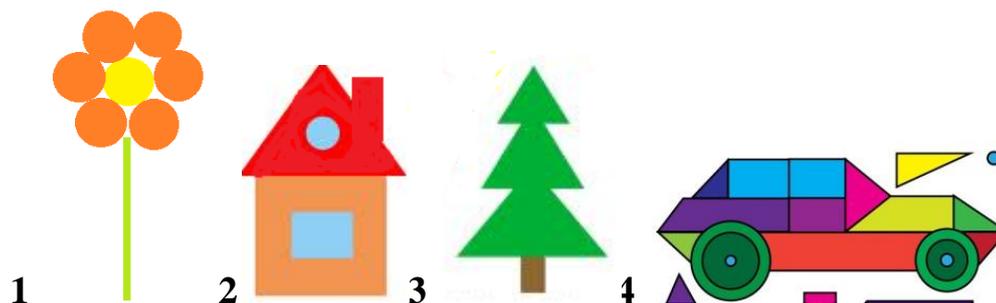


Содержание отчета

- 1 Название работы
- 2 Цель работы
- 3 Технические средства обучения
- 4 Задания (условия задач)
- 5 Порядок выполнения работы
- 6 Ответы на контрольные вопросы
- 7 Вывод

6 Варианты к практической работе:

Нарисовать:





Контрольные вопросы

- 1 Какое пространство имен обеспечивает доступ к функциональным возможностям графического интерфейса?
- 2 Наиболее распространенные классы для работы с графикой.
- 3 Какой класс предоставляет методы рисования на устройстве отображения?
- 4 С чего начинается создание изображения с помощью C#?
- 5 Какой оператор задает «холст» для рисования на форме?
- 6 Какой объект отвечает за место расположения рисунка на «холсте»?
- 7 Какой класс используется для рисования линий, а какой для заливки внутри графических фигур?

Используемая литература

1. Гниденко, И. Г. Технология разработки программного обеспечения : учеб. пособие для СПО / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — М.: Издательство Юрайт, 2017.
2. Шарп Джон Ш26 Microsoft Visual C#. Подробное руководство. 8-е изд. — СПб.: Питер, 2017.
3. Васильев А.Н. Программирование на C# для начинающих. Основные сведения. – Москва: Эксмо, 2018.
4. Васильев А.Н. Программирование на C# для начинающих. Особенности языка. – Москва: Эксмо, 2019.
5. <http://msdn.microsoft.com/ru-ru/library/67ef8sbd.aspx>.