

## Практическая работа 19

### Анимация изображений

**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

#### Краткие теоретические сведения

##### Методы анимации

Проще всего двумерную игру на .NET можно написать, используя стандартную технологию WinForms. Будем отображать графику на всей поверхности формы. Для этого сначала надо получить поверхность рисования GDI+ в виде объекта Graphics.

```
Graphics graphics = this.CreateGraphics();
```

this в данном случае представляет собой объект формы. Сам объект Graphics предоставляет богатый набор методов для рисования линий, кругов, прямоугольников и т.п. Причём вы можете задать, каким цветом рисовать фигуру, заливать ли её цветом всю или только границы фигуры. Также фигуру можно заполнить текстурой (повторяющимся изображением).

Всю эту отрисовку следует выполнять в событии формы OnPaint. Тогда, при перемещении самой формы или изменении её размеров, графика будет перерисовываться автоматически.

Если же вы хотите перерисовывать картинку периодически, например, для создания простейшей анимации в игре, можно инициировать перерисовку по таймеру.

Однако в случае перерисовки по таймеру изображение начинает мерцать. Чтобы избежать мерцания изображения, нужно использовать BufferedGraphicsContext - графический буфер для двойной буферизации. Пример:

```
public Form1()
{
    InitializeComponent();
    Graphics graphics = this.CreateGraphics();
    bufferedGraphicsContext = new BufferedGraphicsContext();
```

```

        bufferedGraphics = bufferedGraphicsContext.Allocate(graphics, new
Rectangle(0, 0, this.Width, this.Height));
    }

```

В конструкторе мы создаём буфер для всей поверхности рисования нашей формы. Далее по таймеру вызываем отрисовку изображения:

```

private void tmr_Tick(object sender, EventArgs e)
{
    // здесь рисуем само изображение
    bufferedGraphics.Graphics.Clear(Color.White);
    engine.Draw(bufferedGraphics.Graphics, this.Size);
    bufferedGraphics.Render();
}

```

Следующий пример кода предназначен для использования совместно с Windows Forms; для него необходим объект [PaintEventArgs](#) и обработчик событий [Paint](#). Этот код выполняет следующие действия:

- Выводит на экран прямоугольник перед применением преобразования поворота (синий прямоугольник).
- Создает матрицу и поворачивает ее на 45 градусов.
- Применяет данное матричное преобразование к прямоугольнику.
- Выводит преобразованный прямоугольник на экран (красный прямоугольник).

Обратите внимание, что красный прямоугольник был повернут вокруг точки экрана с координатами (0, 0).

```

public void RotateExample(PaintEventArgs e)
{
    Pen myPen = new Pen(Color.Blue, 1);
    Pen myPen2 = new Pen(Color.Red, 1);

    // Draw the rectangle to the screen before applying the transform.
    e.Graphics.DrawRectangle(myPen, 150, 50, 200, 100);

    // Create a matrix and rotate it 45 degrees.
    Matrix myMatrix = new Matrix();
    myMatrix.Rotate(45, MatrixOrder.Append);

    // Draw the rectangle to the screen again after applying the

    // transform.
    e.Graphics.Transform = myMatrix;
    e.Graphics.DrawRectangle(myPen2, 150, 50, 200, 100);
}

```

## **Задания**

- 1 Изучить теоретические сведения и задание к работе
- 2 В соответствии с вариантом задания практической работы 18 разработать и отладить программный модуль с анимацией рисунка

## **Порядок выполнения работы**

Создадим новый проект WinForms, добавим pictureBox, таймер и три кнопки, одна запускает анимацию, другая останавливает, третья для завершения работы программы.

...

## **Содержание отчета**

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

## **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>.