**Реферат по лекции «Рисование»**

При создании оконного приложения Windows система вставляет практически весь код сама. В коллбэк функции WndProc продемонстрирован код для выполнения тех или иных действий в зависимости от нажатых клавиш. Обработка представлена с помощью конструкции «switch-case». Например, обработка события WM\_PAINT:

case WM\_PAINT: //рисование главного окна

hdc = BeginPaint(hWnd, &ps);

**GamePaint(hdc,hWnd);**

EndPaint(hWnd, &ps);

break;

Можно заметить, что работа с рисунками требует активации графика (BeginPaint/EndPaint)

Рассмотрим фрагменты кода функции GamePaint:

1. Вывод текста

*TextOut(hDC, 10, 10, TEXT("Oleg German"), 11);*

где 10, 10 – координаты в окне

Text(“…”) – текст для вывода

11 – количество выводимых символов

1. Рисование линии

*LineTo(hDC, 100, 25);*

Линия рисуется с точки курсора до точки, указанной в команде (100, 25)

1. Перевод курсора в новую позицию – *MoveToEx(hDC, 10, 25, NULL);*
2. Рисование прямоугольника

*Rectangle(hDC, 34, 50, 54, 70);*

34, 50 – координаты верхнего левого угла

54, 70 – координаты нижнего правого угла

1. Рисование эллипса

*Ellipse(hDC, 40, 55, 48, 65);*

40, 55 – координаты верхнего левого угла

48, 65 – координаты правого нижнего угла

Это координаты прямоугольника, в который вписывается эллипс

1. Рисование закрашенного прямоугольника

* Создаем кисть красного цвета

*if( !brRed ) brRed = CreateSolidBrush( RGB( 200, 0, 0 ) );*

* Рисуем красный прямоугольник

*FillRect( hDC, &rt, brRed );*

**Отображение рисунков:**

Считывание рисунка из файла

*hbitmap = (HBITMAP)LoadImage(hInst, L"c:\\work\\Figs\\1.bmp", IMAGE\_BITMAP, 0, 0, LR\_LOADFROMFILE);*

Создаем графический контекст для отображения рисунков

*hdcMem = CreateCompatibleDC(hdc);*

Получаем объект bitmap

*GetObject(hbitmap, sizeof(bitmap), &bitmap);*

Рисуем

*BitBlt(hdc, 0, 0, bitmap.bmWidth, bitmap.bmHeight, hdcMem, 0, 0, SRCCOPY);*

Можно инициировать рисование, чтобы обновить внешний вид окна, как приложением, так и ОС (при растяжке окна). При этом ОС отправляет окну сообщение WM\_PAINT.

**Область обновления –** та часть окна, которая должна быть нарисована.

**UpdateWindow()** – функция для обновления содержимого окна, которая посылает сообщение WM\_PAINT в оконную процедуру.

**InvalidateRect()** – функция для обновления содержимого окна, которая посылает сообщение перерисовки в оконную процедуру без задержки, т.е. минуя очередь.

Программа ответственная за заливку клиентской области. Окружающая рамка автоматически окрашивается ОС. Необходимо очистить область обновления после того, как закончится рисование клиентской области.

Область обновления изменяется, если пользователь растягивает окно.

Canvas позволяет рисовать сценарии, которые включают в себя прямоугольники, контуры, линии, кривые Безье, заливки, дуги и квадратные кривые.

IE 9 поддерживает API-интерфейсы контекста Canvas:

* свойство canvas
* методы состояния
* свойства компоновки
* свойства цвета и стиля
* атрибуты линии
* теневые атрибуты
* прямоугольные методы
* методы сложных фигур
* метод изображений

и другие API-интерфейсы