МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

## «БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

РГЗ

по дисциплине: Архитектура вычислительных систем

тема: «WinApi для отрисовки простых фигур в ассемблер»

# Выполнил: ст. группы ПВ-223

# Дмитриев Андрей Александрович

Проверил:

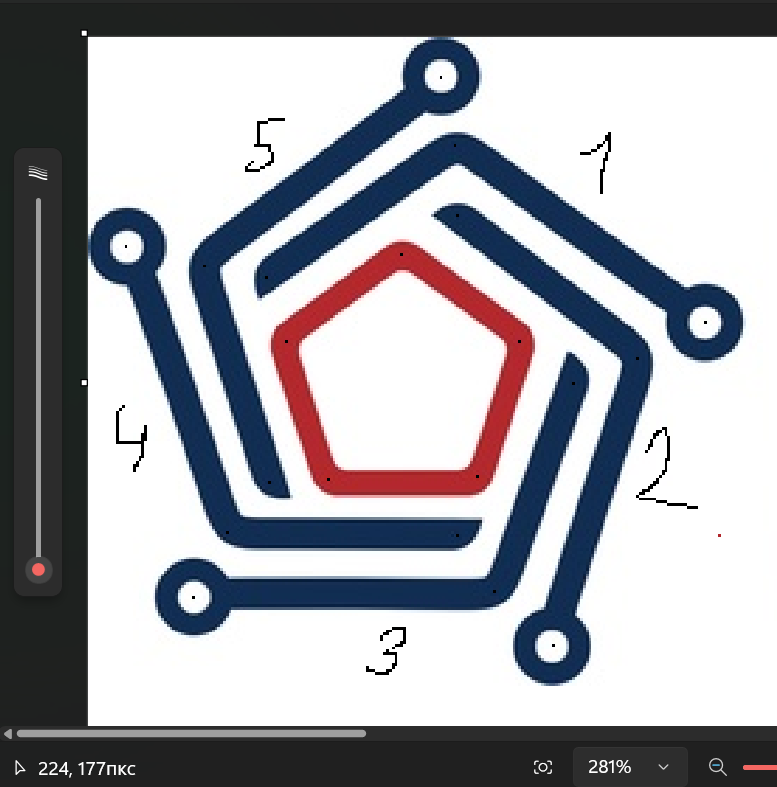
# Осипов Олег Васильевич

Белгород 2024 г.

# Вариант 2

Ход работы:

Для удобства снятия точек с эмблемы использовал paint, который отображал положение указателя мыши. Цвет был снят с помощью инструмента «палитра».



Для построения использовалась процедура Polyline для рисования линий фигур, для кружочков использовался приём с утолщением кисти.

Код программ:

.386

.model flat,stdcall

option casemap:none

include C:\masm32\include\windows.inc

include C:\masm32\include\user32.inc

include C:\masm32\include\kernel32.inc

include C:\masm32\include\gdi32.inc

include C:\masm32\include\msvcrt.inc

includelib C:\masm32\lib\msvcrt.lib

includelib C:\masm32\lib\user32.lib

includelib C:\masm32\lib\kernel32.lib

includelib C:\masm32\lib\gdi32.lib

WinMain proto :DWORD,:DWORD,:DWORD,:DWORD

.data

ClassName db "SimpleWinClass",0

AppName db "IITUS",0

hInstance HINSTANCE ?

hPen HPEN ?

.code

start:

    invoke GetModuleHandle, NULL

    mov hInstance, eax

    invoke WinMain, hInstance, NULL, NULL, SW\_SHOWDEFAULT

    invoke ExitProcess, eax

WinMain proc hInst:HINSTANCE, hPrevInst:HINSTANCE, CmdLine:LPSTR, CmdShow:DWORD

    LOCAL wc:WNDCLASSEX

    LOCAL msg:MSG

    LOCAL hwnd:HWND

    mov wc.cbSize, SIZEOF WNDCLASSEX

    mov wc.style, CS\_HREDRAW or CS\_VREDRAW

    mov wc.lpfnWndProc, OFFSET WndProc

    mov wc.cbClsExtra, NULL

    mov wc.cbWndExtra, NULL

    push hInst

    pop wc.hInstance

    mov wc.hbrBackground, COLOR\_WINDOW+1

    mov wc.lpszMenuName, NULL

    mov wc.lpszClassName, OFFSET ClassName

    invoke LoadIcon, NULL, IDI\_APPLICATION

    mov wc.hIcon, eax

    mov wc.hIconSm, eax

    invoke LoadCursor, NULL, IDC\_ARROW

    mov wc.hCursor, eax

    invoke RegisterClassEx, addr wc

    invoke CreateWindowEx, NULL, ADDR ClassName, ADDR AppName, WS\_OVERLAPPEDWINDOW, CW\_USEDEFAULT, CW\_USEDEFAULT, 250, 300, NULL, NULL, hInst, NULL

    mov hwnd, eax

    invoke ShowWindow, hwnd, CmdShow

    invoke UpdateWindow, hwnd

    .WHILE TRUE

        invoke GetMessage, ADDR msg, NULL, 0, 0

        .BREAK .IF (!eax)

        invoke TranslateMessage, ADDR msg

        invoke DispatchMessage, ADDR msg

    .ENDW

    mov eax, msg.wParam

    ret

WinMain endp

WndProc proc hWnd:HWND, uMsg:UINT, wParam:WPARAM, lParam:LPARAM

    LOCAL hdc:HDC

    LOCAL ps:PAINTSTRUCT

    LOCAL hMainPen:HPEN

    LOCAL hSubPen:HPEN

    LOCAL points[6]:POINT

    .IF uMsg == WM\_DESTROY

        invoke PostQuitMessage, 0

    .ELSEIF uMsg == WM\_PAINT

        invoke BeginPaint, hWnd, ADDR ps

        mov hdc, eax

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

        ; пятиугольник

        mov points[0 \* TYPE POINT].x, 110

        mov points[0 \* TYPE POINT].y, 80

        mov points[1 \* TYPE POINT].x, 150

        mov points[1 \* TYPE POINT].y, 110

        mov points[2 \* TYPE POINT].x, 135

        mov points[2 \* TYPE POINT].y, 155

        mov points[3 \* TYPE POINT].x, 85

        mov points[3 \* TYPE POINT].y, 155

        mov points[4 \* TYPE POINT].x, 70

        mov points[4 \* TYPE POINT].y, 110

        mov points[5 \* TYPE POINT].x, 110

        mov points[5 \* TYPE POINT].y, 80

        invoke CreatePen, PS\_SOLID, 10, 02B26B1h

        mov hMainPen, eax

        invoke SelectObject, hdc, hMainPen

        invoke Polyline, hdc, ADDR points, 6

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

        ; линии

        invoke CreatePen, PS\_SOLID, 10, 0563012h

        mov hMainPen, eax

        invoke SelectObject, hdc, hMainPen

        ; 1

        mov points[0 \* TYPE POINT].x, 130

        mov points[0 \* TYPE POINT].y, 64

        mov points[1 \* TYPE POINT].x, 197

        mov points[1 \* TYPE POINT].y, 113

        mov points[2 \* TYPE POINT].x, 163

        mov points[2 \* TYPE POINT].y, 216

        invoke Polyline, hdc, ADDR points, 3

        ; 2

        mov points[0 \* TYPE POINT].x, 172

        mov points[0 \* TYPE POINT].y, 120

        mov points[1 \* TYPE POINT].x, 145

        mov points[1 \* TYPE POINT].y, 197

        mov points[2 \* TYPE POINT].x, 37

        mov points[2 \* TYPE POINT].y, 197

        invoke Polyline, hdc, ADDR points, 3

        ; 3

        mov points[0 \* TYPE POINT].x, 130

        mov points[0 \* TYPE POINT].y, 176

        mov points[1 \* TYPE POINT].x, 49

        mov points[1 \* TYPE POINT].y, 176

        mov points[2 \* TYPE POINT].x, 13

        mov points[2 \* TYPE POINT].y, 74

        invoke Polyline, hdc, ADDR points, 3

        ; 4

        mov points[0 \* TYPE POINT].x, 63

        mov points[0 \* TYPE POINT].y, 156

        mov points[1 \* TYPE POINT].x, 38

        mov points[1 \* TYPE POINT].y, 81

        mov points[2 \* TYPE POINT].x, 125

        mov points[2 \* TYPE POINT].y, 14

        invoke Polyline, hdc, ADDR points, 3

        ; 5

        mov points[0 \* TYPE POINT].x, 63

        mov points[0 \* TYPE POINT].y, 88

        mov points[1 \* TYPE POINT].x, 131

        mov points[1 \* TYPE POINT].y, 38

        mov points[2 \* TYPE POINT].x, 219

        mov points[2 \* TYPE POINT].y, 101

        invoke Polyline, hdc, ADDR points, 3

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

        ; петельки

        invoke CreatePen, PS\_SOLID, 24, 0563012h

        mov hMainPen, eax

        invoke CreatePen, PS\_SOLID, 11, 0FFFFFFh

        mov hSubPen, eax

        ; 1

        mov points[0 \* TYPE POINT].x, 163

        mov points[0 \* TYPE POINT].y, 216

        mov eax, points[0 \* TYPE POINT].x

        mov points[1 \* TYPE POINT].x, eax

        mov eax, points[0 \* TYPE POINT].y

        mov points[1 \* TYPE POINT].y, eax

        invoke SelectObject, hdc, hMainPen

        invoke Polyline, hdc, ADDR points, 2

        invoke SelectObject, hdc, hSubPen

        invoke Polyline, hdc, ADDR points, 2

        ; 2

        mov points[0 \* TYPE POINT].x, 37

        mov points[0 \* TYPE POINT].y, 197

        mov eax, points[0 \* TYPE POINT].x

        mov points[1 \* TYPE POINT].x, eax

        mov eax, points[0 \* TYPE POINT].y

        mov points[1 \* TYPE POINT].y, eax

        invoke SelectObject, hdc, hMainPen

        invoke Polyline, hdc, ADDR points, 2

        invoke SelectObject, hdc, hSubPen

        invoke Polyline, hdc, ADDR points, 2

        ; 3

        mov points[0 \* TYPE POINT].x, 13

        mov points[0 \* TYPE POINT].y, 74

        mov eax, points[0 \* TYPE POINT].x

        mov points[1 \* TYPE POINT].x, eax

        mov eax, points[0 \* TYPE POINT].y

        mov points[1 \* TYPE POINT].y, eax

        invoke SelectObject, hdc, hMainPen

        invoke Polyline, hdc, ADDR points, 2

        invoke SelectObject, hdc, hSubPen

        invoke Polyline, hdc, ADDR points, 2

        ; 4

        mov points[0 \* TYPE POINT].x, 125

        mov points[0 \* TYPE POINT].y, 14

        mov eax, points[0 \* TYPE POINT].x

        mov points[1 \* TYPE POINT].x, eax

        mov eax, points[0 \* TYPE POINT].y

        mov points[1 \* TYPE POINT].y, eax

        invoke SelectObject, hdc, hMainPen

        invoke Polyline, hdc, ADDR points, 2

        invoke SelectObject, hdc, hSubPen

        invoke Polyline, hdc, ADDR points, 2

        ; 5

        mov points[0 \* TYPE POINT].x, 219

        mov points[0 \* TYPE POINT].y, 101

        mov eax, points[0 \* TYPE POINT].x

        mov points[1 \* TYPE POINT].x, eax

        mov eax, points[0 \* TYPE POINT].y

        mov points[1 \* TYPE POINT].y, eax

        invoke SelectObject, hdc, hMainPen

        invoke Polyline, hdc, ADDR points, 2

        invoke SelectObject, hdc, hSubPen

        invoke Polyline, hdc, ADDR points, 2

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

        ; конец отрисовки

        invoke DeleteObject, hMainPen

        invoke DeleteObject, hSubPen

        invoke EndPaint, hWnd, ADDR ps

    .ELSE

        invoke DefWindowProc, hWnd, uMsg, wParam, lParam

        ret

    .ENDIF

    xor eax, eax

    ret

WndProc endp

end start

Вывод программы:



**Вывод:** Входе лабораторной работы изучены команды поразрядной обработки данных.