ФГБОУ ВО

Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет) «МГТУ им. Н.Э. Баумана»

Факультет ИУ «Информатика и системы управления»

Кафедра ИУ-3 «Информационные системы и телекоммуникации»

Отчет по лабораторной работе 7

**«Win32API»**

По курсу «Информатика»

Выполнил: Еньшин Б. А.

Группа ИУ3-12

Вариант 9

Проверил: Видьманов Д. А.

Москва 2019

**Цель работы:**

Ознакомиться с основами построения классического оконного приложения Windows. Узнать о правильных способах создать окно. Научиться обрабатывать события в этом окне. Научиться рисовать в окне графику. Создать собственное оконное приложение Win32API по заданию.

**Постановка задания:**

Создать проект Win32, в окне нарисовать взлетающую ракету, у которой корпус – зеленого цвета, иллюминаторы – синие, пламя - оранжевое. С помощью элемента управления – кнопки1 – отображать рисунок на экране, с помощью кнопки2 – очищать окно от рисунка, с помощью кнопки3 – появляется надпись фамилии, созданная TextOut, того, кто сделал эту работу.

**Ниже приведен код программы lab7\_source.cpp**

#include <windows.h>

#include <time.h>

LRESULT CALLBACK WndProc(HWND, UINT, WPARAM, LPARAM);

HINSTANCE hInst;

BOOL Line(HDC hdc, int x1, int y1, int x2, int y2)

{

MoveToEx(hdc, x1, y1, NULL); //сделать текущими координаты x1, y1

return LineTo(hdc, x2, y2);

}

int WINAPI WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance,

PSTR szCmdLine, int iCmdShow)

{

LPCSTR szAppName = "HelloWin";

HWND hwnd;

MSG msg;

WNDCLASSEX wndclass;

wndclass.cbSize = sizeof(wndclass);

wndclass.style = CS\_HREDRAW | CS\_VREDRAW;

wndclass.lpfnWndProc = WndProc;

wndclass.cbClsExtra = 0;

wndclass.cbWndExtra = 0;

wndclass.hInstance = hInstance;

wndclass.hIcon = LoadIcon(NULL, IDI\_APPLICATION);

wndclass.hCursor = LoadCursor(NULL, IDC\_ARROW);

wndclass.hbrBackground = (HBRUSH)GetStockObject(WHITE\_BRUSH);

wndclass.lpszMenuName = NULL;

wndclass.lpszClassName = L"HelloWin";

wndclass.hIconSm = LoadIcon(NULL, IDI\_APPLICATION);

RegisterClassEx(&wndclass);

hwnd = CreateWindowA(szAppName,

"Ракета",

WS\_OVERLAPPEDWINDOW,

CW\_USEDEFAULT,

CW\_USEDEFAULT,

CW\_USEDEFAULT,

CW\_USEDEFAULT,

NULL,

NULL,

hInstance,

NULL);

ShowWindow(hwnd, iCmdShow);

UpdateWindow(hwnd);

while (GetMessage(&msg, NULL, 0, 0))

{

TranslateMessage(&msg);

DispatchMessage(&msg);

}

return msg.wParam;

}

#define BTN\_WIDTH 150

#define BTN\_HEIGHT 20

LRESULT CALLBACK WndProc(HWND hwnd, UINT iMsg, WPARAM wParam, LPARAM lParam)

{

HDC hdc, hdc1, hdc2;

PAINTSTRUCT ps;

RECT rect;

static HWND hwndButton1, hwndButton2, hwndButton3;

static char txt[32] = "";

int loword, hiword;

static int fpic = 0;

static HPEN hpen;

static HBRUSH hbrush, hbrush1, hbrush2, hbrush3, hbrush4;

switch (iMsg)

{

case WM\_CREATE:

srand(time(0));

hdc = GetDC(hwnd);

hdc1 = GetDC(hwnd);

hdc2 = GetDC(hwnd);

hwndButton1 = CreateWindowA("button", "Нарисовать ракету",

WS\_CHILD | WS\_VISIBLE | BS\_PUSHBUTTON,

100, 50, BTN\_WIDTH, BTN\_HEIGHT,

hwnd, (HMENU)1, hInst, NULL);

hwndButton2 = CreateWindowA("button", "Стереть рисунок",

WS\_CHILD | WS\_VISIBLE | BS\_PUSHBUTTON,

100, 100, BTN\_WIDTH, BTN\_HEIGHT,

hwnd, (HMENU)2, hInst, NULL);

hwndButton3 = CreateWindowA("button", "Автор",

WS\_CHILD | WS\_VISIBLE | BS\_PUSHBUTTON,

100, 150, BTN\_WIDTH, BTN\_HEIGHT,

hwnd, (HMENU)3, hInst, NULL);

return 0;

case WM\_COMMAND:

hbrush = CreateSolidBrush(RGB(0, 64, 0));

hbrush1 = CreateSolidBrush(RGB(255, 255, 255));

hbrush2 = CreateSolidBrush(RGB(0, 52, 0));

hbrush3 = CreateSolidBrush(RGB(255, 100, 0));

hbrush4 = CreateSolidBrush(RGB(0, 0, 255));

loword = LOWORD(wParam);

hiword = HIWORD(wParam);

switch (loword) {

case 1:strcpy\_s(txt, "Ракетаааа"); fpic = 1; break;

case 2:strcpy\_s(txt, ""); fpic = 2; break;

case 3:strcpy\_s(txt, "Ф.И.О."); fpic = 3; break;

}

InvalidateRect(hwnd, NULL, TRUE);

return 0;

case WM\_PAINT:

hdc = BeginPaint(hwnd, &ps);

hdc1 = BeginPaint(hwnd, &ps);

hdc2 = BeginPaint(hwnd, &ps);

GetClientRect(hwnd, &rect);

TextOutA(hdc, 100, 300, txt, strlen(txt));

SelectObject(hdc, hbrush);

TextOutA(hdc1, 100, 300, txt, strlen(txt));

SelectObject(hdc1, hbrush1);

switch (fpic) {

case 1:

HRGN Rgn2, Rgn3, Rgn4\_l, Rgn4\_r;

Rectangle(hdc, 500, 200, 600, 450);

POINT ptArray1[3], ptArray2[3], ptArray3\_l[4], ptArray3\_r[4];

ptArray2[0].x = 500; ptArray2[0].y = 200;

ptArray2[1].x = 550; ptArray2[1].y = 100;

ptArray2[2].x = 600; ptArray2[2].y = 200;

Rgn2 = CreatePolygonRgn(ptArray2, sizeof(ptArray2) / sizeof(POINT), ALTERNATE);

FillRgn(hdc, Rgn2, hbrush2);

ptArray1[0].x = 490; ptArray1[0].y = 450;

ptArray1[1].x = 550; ptArray1[1].y = 550;

ptArray1[2].x = 610; ptArray1[2].y = 450;

Rgn3 = CreatePolygonRgn(ptArray1, sizeof(ptArray1) / sizeof(POINT), ALTERNATE);

FillRgn(hdc, Rgn3, hbrush3);

ptArray3\_l[0].x = 500; ptArray3\_l[0].y = 400;

ptArray3\_l[1].x = 470; ptArray3\_l[1].y = 430;

ptArray3\_l[2].x = 470; ptArray3\_l[2].y = 500;

ptArray3\_l[3].x = 500; ptArray3\_l[3].y = 470;

Rgn4\_l = CreatePolygonRgn(ptArray3\_l, sizeof(ptArray3\_l) / sizeof(POINT), ALTERNATE);

FillRgn(hdc, Rgn4\_l, hbrush2);

ptArray3\_r[0].x = 600; ptArray3\_r[0].y = 400;

ptArray3\_r[1].x = 630; ptArray3\_r[1].y = 430;

ptArray3\_r[2].x = 630; ptArray3\_r[2].y = 500;

ptArray3\_r[3].x = 600; ptArray3\_r[3].y = 470;

Rgn4\_r = CreatePolygonRgn(ptArray3\_r, sizeof(ptArray3\_r) / sizeof(POINT), ALTERNATE);

FillRgn(hdc, Rgn4\_r, hbrush2);

HRGN Rgn\_illume;

Rgn\_illume = CreateEllipticRgn(530, 240, 570, 280);

FillRgn(hdc, Rgn\_illume, hbrush4);

break;

case 2:

Rectangle(hdc1, 300, 100, 800, 400);

break;

case 3:

TextOutA(hdc, 100, 300, "Еньшин Богдан ИУ3-22Б", strlen("Еньшин Богдан ИУ3-22Б"));

break;

}

EndPaint(hwnd, &ps);

return 0;

case WM\_DESTROY:

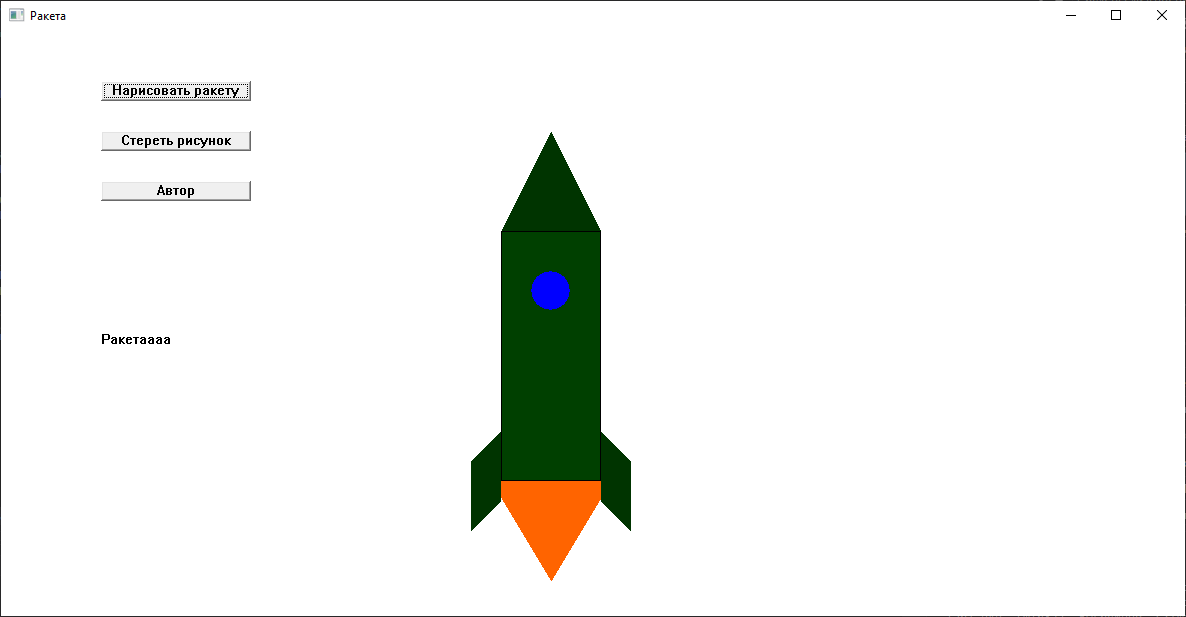
PostQuitMessage(0);

return 0;

}

return DefWindowProc(hwnd, iMsg, wParam, lParam);

}

**Скриншоты приложения:**  

**Вывод:**

В ходе данной лабораторной работы я получил навыки разработки классических оконных приложений на Win32API. Я научился создавать окно и рисовать графику в нем. Также я научился обрабатывать пользовательские ситуации, такие как нажатие интерактивной кнопки в окне. Данные знания были подкреплены решением поставленной задачи на создание оконного приложения на Win32API.