Отчет по лабораторной работе № 4

«Использование окон произвольного размера»

7 вариант

Выполнил: студент группы P3317

Плюхин Д.А.

Преподаватель: Зинчик Александр Адольфович

# **Задание**

# Написать программу, выполняющую вертикальное перемещение трапеции по рабочему столу, добавить возможность выбора скорости перемещения. При перемещении фигура не должна мерцать. При достижении края экрана показывать «появляющейся» с другой стороны.

# **Исходный код**

## Создание окна в виде трапеции

HBITMAP hbmp = (HBITMAP)LoadImage(NULL, L"C:\\Users\\Zerbs\\Desktop\\interfaces\\lab\_4\\star.bmp", IMAGE\_BITMAP, 0, 0, LR\_LOADFROMFILE | LR\_CREATEDIBSECTION);

BITMAP bm;

GetObject(hbmp, sizeof(bm), &bm);

BYTE \*pMaskBits = (BYTE \*)bm.bmBits;

int pixel;

int xStart = -1;

HRGN hRgn = 0;

for (int i = 0; i < bm.bmHeight; i++) {

for (int j = 0; j < bm.bmWidth; j++) {

pixel = (pMaskBits[i \* bm.bmWidthBytes + j]);

if (pixel != 0) {

if (xStart == -1) xStart = j;

}

else if (xStart != -1) {

if (hRgn == 0) {

hRgn = CreateRectRgn(xStart, bm.bmHeight - i - 1, j, bm.bmHeight - i);

}

else {

CombineRgn(hRgn, hRgn, CreateRectRgn(xStart,

bm.bmHeight - i - 1, j, bm.bmHeight - i), RGN\_OR);

xStart = -1;

}

}

if (j == bm.bmWidth - 1 && xStart != -1) {

if (hRgn == 0)

hRgn = CreateRectRgn(xStart, bm.bmHeight - i - 1, j + 1, bm.bmHeight - i);

else CombineRgn(hRgn, hRgn, CreateRectRgn(xStart, bm.bmHeight - i - 1, j + 1, bm.bmHeight - i), RGN\_OR);

xStart = -1;

}

}

}

SetWindowRgn(hWnd, hRgn, true);

ShowWindow(hWnd, nCmdShow);

UpdateWindow(hWnd);

|  |  |
| --- | --- |
| Исходный файл (.bmp) | Полученное окно |
| C:\Users\Zerbs\AppData\Local\Microsoft\Windows\INetCache\Content.Word\star.bmp |  |

## Обеспечение перемещения трапеции по рабочему столу

void UpdateDesktopSize(int& horizontal, int& vertical, HWND hWnd) {

RECT desktop;

const HWND hDesktop = GetDesktopWindow();

GetWindowRect(hDesktop, &desktop);

horizontal = desktop.right;

vertical = desktop.bottom;

bmpOffset.x = (int)(horizontal / 2) - (int)(bmpSize.cx / 2);

UpdateWindowPosition(hWnd);

}

void UpdateWindowPosition(HWND hWnd) {

SetWindowPos

(

hWnd,

HWND\_TOP,

bmpOffset.x,

bmpOffset.y,

bmpSize.cx,

bmpSize.cy,

0

);

}

void CALLBACK Do\_Step(\_In\_ HWND hWnd, \_In\_ UINT uMsg, \_In\_ UINT\_PTR idEvent, \_In\_ DWORD dwTime) {

bmpOffset.y = bmpOffset.y + step;

if (bmpOffset.y > screenHeight) {

bmpOffset.y = -bmpSize.cy;

}

else if (bmpOffset.y < -bmpSize.cy) {

bmpOffset.y = screenHeight;

}

UpdateWindowPosition(hWnd);

}

SetTimer(hWnd, 1, 10, (TIMERPROC)Do\_Step);

## Обеспечение регулирования скорости перемещения

HHOOK wheelRotatedHook;

LRESULT \_\_declspec(dllexport)\_\_stdcall CALLBACK WheelProc(int nCode, WPARAM wParam, LPARAM lParam) {

if (nCode >= 0) {

if (wParam == WM\_MOUSEWHEEL) {

MSLLHOOKSTRUCT \*pMhs = (MSLLHOOKSTRUCT \*)lParam;

short zDelta = HIWORD(pMhs->mouseData);

if (zDelta > 0) {

step++;

}

else {

step--;

}

}

}

return CallNextHookEx(wheelRotatedHook, nCode, wParam, lParam);

}

wheelRotatedHook = SetWindowsHookEx(WH\_MOUSE\_LL, (HOOKPROC)WheelProc, 0, 0);

# **Вывод**

Таким образом, WINAPI позволяет достаточно просто и эффективно осуществлять создание и использование окон произвольной формы, а также добавление функций обработки ввода пользователя без использования сторонних библиотек.