Министерство науки и образования Российской Федерации

Пензенский государственный университет

Кафедра «Системы автоматизированного проектирования»

**Отчет**

По лабораторной работе №2

По курсу «Операционные системы реального времени»

На тему «Передача данных с использованием почтовых ящиков и ждущих таймеров, перехват и фильтрация информации с использованием системных перехватчиков»

Выполнил студент гр.20ВВС1

Горбунов Н.А.

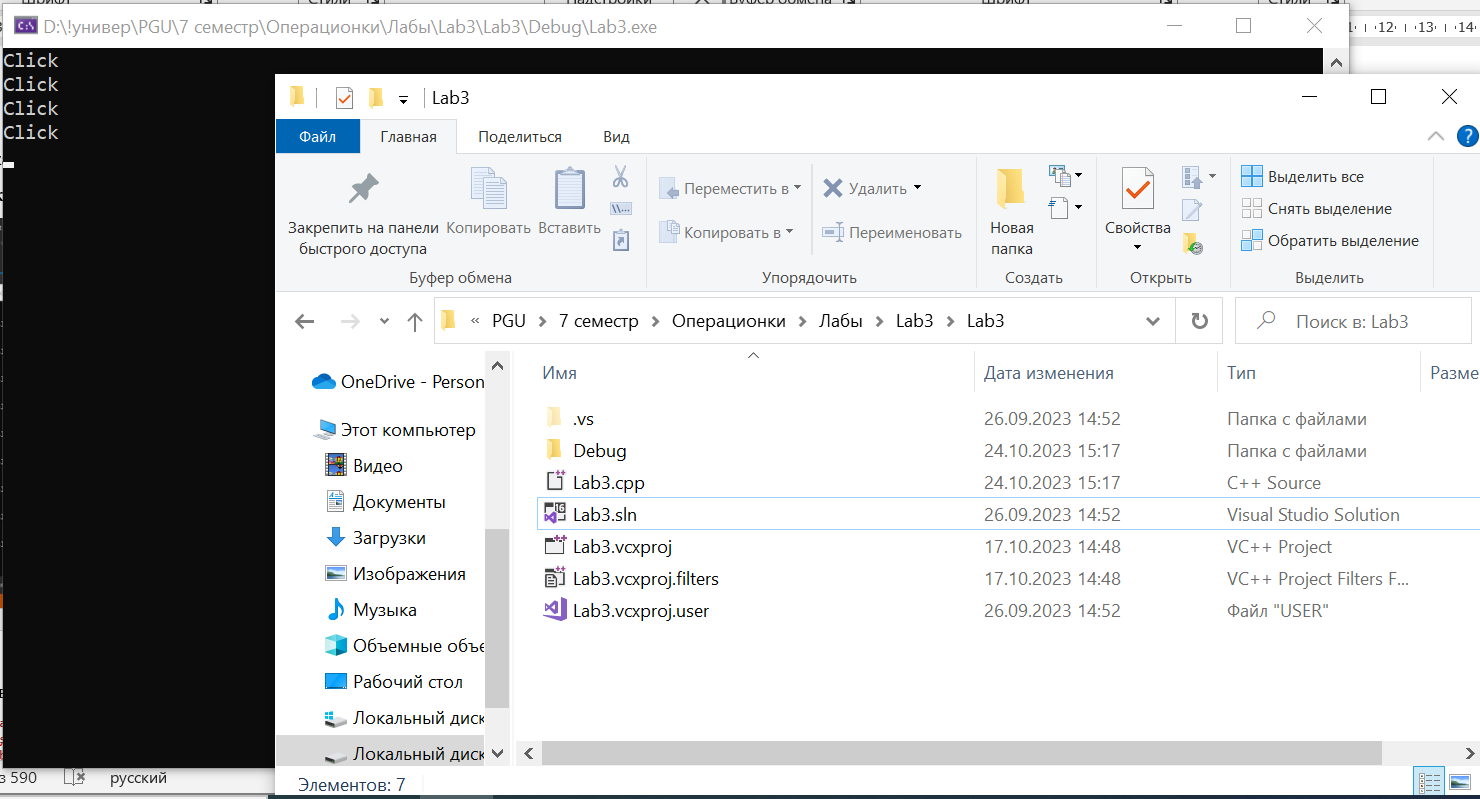
Проверил:

Гудков П.А.

Пенза, 2023

Задание 2 вариант 22:

**Цель:** разработать программу, выводящую на экран текстовое сообщение при нажатии пользователем левой кнопки мыши в заголовке окна.



По нажатии на заголовок любого окна выводится в консоль “Click”

**Листинг:**

#pragma comment( lib, "user32.lib" )

#include <strsafe.h>

#include <windows.h>

#include <stdio.h>

#include <iostream>

HHOOK hMouseHook;

UINT a1;

bool busy = false;

#pragma comment( lib, "user32.lib")

#pragma comment( lib, "gdi32.lib")

#define NUMHOOKS 7

// Global variables

typedef struct \_MYHOOKDATA

{

int nType;

HOOKPROC hkprc;

HHOOK hhook;

} MYHOOKDATA;

MYHOOKDATA myhookdata[NUMHOOKS];

HWND gh\_hwndMain;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Подключаемая процедура WH\_CALLWNDPROC

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

// The callback function

LRESULT CALLBACK CallWndProc(int nCode, WPARAM wParam, LPARAM lParam) {

if (nCode == HC\_ACTION)

{

if (wParam > 0)

{

std::cout << "sent by the current thread wParam=" << wParam;

}

CWPSTRUCT\* s = (CWPSTRUCT\*)lParam;

}

//This is a must

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

}

LPWSTR windowName;

bool Click = false;

LRESULT CALLBACK mouseProc(int nCode, WPARAM wParam, LPARAM lParam)

{

//printf("mouseProc");

MOUSEHOOKSTRUCT\* pMouseStruct = (MOUSEHOOKSTRUCT\*)lParam;

UINT HitTestCode;

HitTestCode = ((MOUSEHOOKSTRUCT\*)lParam)->wHitTestCode;

//std::cout << HitTestCode;

if (pMouseStruct != NULL)

{

POINT p;

GetCursorPos(&p);

HWND window = WindowFromPoint(p);

RECT rect;

if (GetWindowRect(window, &rect))

{

int width = rect.right - rect.left;

int height = rect.bottom - rect.top;

if (p.y >= rect.top && p.y <= rect.top + 30) {

if (wParam == WM\_LBUTTONDOWN) {

printf("Click\n");

}

}

}

}

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

}

DWORD WINAPI MyMouseLogger()

{

HINSTANCE hInstance = GetModuleHandle(NULL);

HWND hwnd = FindWindow(NULL, UNIQUE\_NAME);

DWORD pid = NULL; // If we dont know -> NULL

DWORD tid = GetWindowThreadProcessId(hwnd, &pid);

hMouseHook = SetWindowsHookEx(WH\_MOUSE\_LL, mouseProc, hInstance, NULL);

return 0;

}

/\* Variable to store the HANDLE to the hook. Don't declare it anywhere else then globally

or you will get problems since every function uses this variable. \*/

HHOOK \_hook;

// the dll instance

HINSTANCE \_dllInstance;

int main() {

MyMouseLogger();

MSG msg;

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

TranslateMessage(&msg);

DispatchMessage(&msg);

}

}

**Вывод:** разработал программу, выводящую на экран текстовое сообщение при нажатии пользователем левой кнопки мыши в заголовке окна.