

Министерство образования Республики Беларусь
Белорусский государственный университет информатики и радиоэлектроники
Кафедра инженерной психологии и эргономики

Отчет по лабораторной работе №11-12
«Обмен сообщениями между окнами»
Вариант №7

Выполнила:

Студент гр.910102
Воспьяков И.И.

Проверила:

Киринович И.Ф.

Минск 2022

Цель: Реализация механизма обмена сообщениями в ОС Windows.

Задачи:

1. Изучение теоретического материала по обмену сообщениями для управления окнами.
2. Составление алгоритма программы.
3. Программная реализация.

Задание: Разработать программу, которая создает окно. Заккрытие окна должно выполняться щелчком кнопки по значку разворачивания окна, при этом должно создаваться новое окно с таким же стилем. Однократный щелчок кнопки по значку закрытия не приводит к выполнению этого действия. Двойной щелчок по этому значку завершает работу программы.

Листинг кода

```
#include <windows.h>

// объявление функций
LRESULT CALLBACK WndProc(HWND, UINT, WPARAM, LPARAM);
ATOM RegMyWindowClass(HINSTANCE, LPCTSTR);
WINDOWPLACEMENT wnd1;

HWND hWnd;

int check = 0, check1 = 0;

// функция вхождений программы WinMain
int APIENTRY WinMain(HINSTANCE hInstance,
                    HINSTANCE hPrevInstance,
                    LPSTR lpCmdLine,
                    int nCmdShow)
{
    // имя будущего класса
    LPCTSTR lpzClass = TEXT("My Window Class!");
    // регистрация класса
    if (!RegMyWindowClass(hInstance, lpzClass))
        return 1;
    // вычисление координат центра экрана
    RECT screen_rect;
```

```

    GetWindowRect(GetDesktopWindow(), &screen_rect); // разрешение
экрана
    int x = screen_rect.right / 2 - 150;
    int y = screen_rect.bottom / 2 - 75;

    // создание диалогового окна
    if (check1 == 0) {
        hWnd = CreateWindow(lpzClass, TEXT("Dialog Window"),
            WS_OVERLAPPEDWINDOW | WS_VISIBLE, x, y, 300, 250,
NULL, NULL,
            hInstance, NULL);
        check1++;
    }
    // если окно не создано, описатель будет равен 0
    if (!hWnd) return 2;

    MSG msg = { 0 }; // структура сообщения
    int iGetOk = 0;

    while ((iGetOk = GetMessage(&msg, NULL, 0, 0)) != 0) // цикл
сообщений
    {
        if (iGetOk == -1) return 3; // если GetMessage вернул
ошибку - выход
        TranslateMessage(&msg);
        DispatchMessage(&msg);
        GetWindowPlacement(hWnd, &wnd1);

        if (wnd1.showCmd != 0 && check == 0) {
            hWnd = CreateWindow(lpzClass, TEXT("Dialog Window"),
                WS_OVERLAPPEDWINDOW | WS_VISIBLE, x, y, 300, 300,
NULL, NULL,
                hInstance, NULL);
            check++;
        }
    }
    return msg.wParam; // возвращаем код завершения программы
}

ATOM RegMyWindowClass(HINSTANCE hInst, LPCTSTR lpzClassName)
{
    WNDCLASS wcWindowClass = { 0 };
    // адрес ф-ции обработки сообщений
    wcWindowClass.lpfnWndProc = (WNDPROC)WndProc;
    // стиль окна
    wcWindowClass.style = CS_HREDRAW | CS_VREDRAW;
    // дескриптор экземпляра приложения
    wcWindowClass.hInstance = hInst;
    // название класса

```

```

wcWindowClass.lpszClassName = lpzClassName;
// загрузка курсора
wcWindowClass.hCursor = LoadCursor(NULL, IDC_ARROW);
// загрузка цвета окон
wcWindowClass.hbrBackground = (HBRUSH)COLOR_APPWORKSPACE;

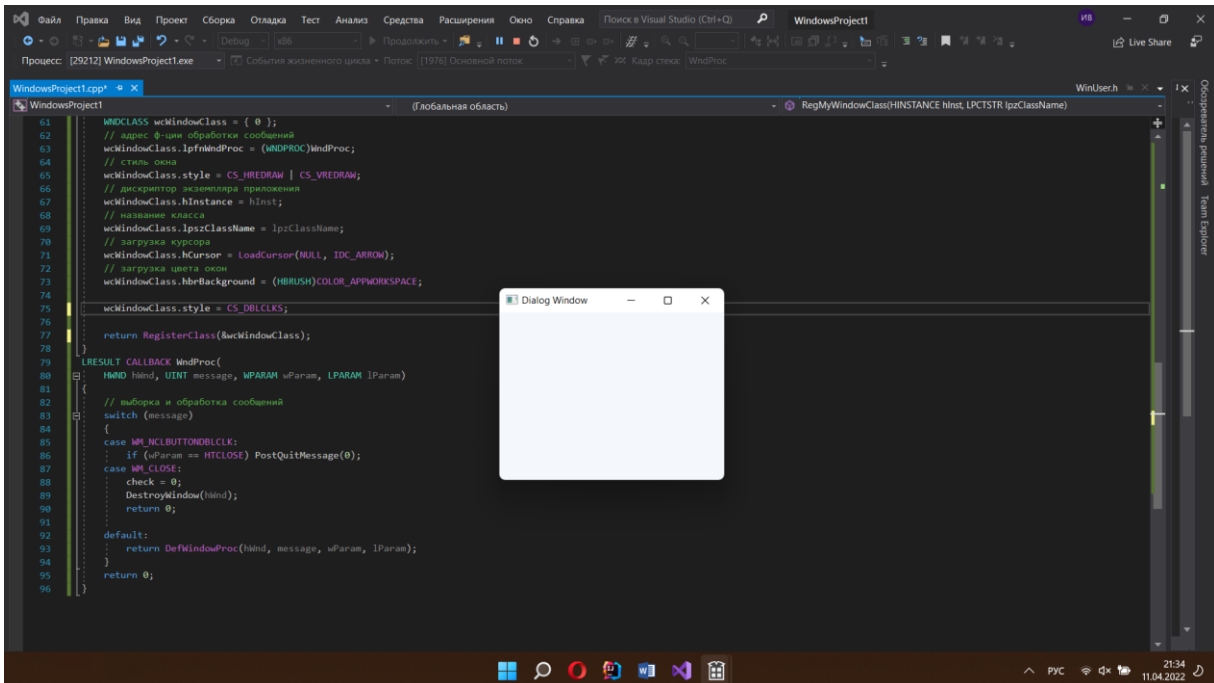
wcWindowClass.style = CS_DBLCLKS;

return RegisterClass(&wcWindowClass); // регистрация класса
}
LRESULT CALLBACK WndProc(
    HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)
{
    // выборка и обработка сообщений
    switch (message)
    {
        case WM_NCLBUTTONDBLCLK:
            if (wParam == HTCLOSE) PostQuitMessage(0);
        case WM_CLOSE:
            check = 0;
            DestroyWindow(hWnd);
            return 0;

        default:
            return DefWindowProc(hWnd, message, wParam, lParam);
    }
    return 0;
}

```

Выполнение программы



При нажатии на значок разворачивания окна текущее окно закрывается и открывается новое. Двойной щелчок по этому значку завершает работу программы.

Вывод: Были изучены теоритические материалы по созданию окон в ОС Windows и выполнено практическое задание, подкрепляющее полученные знания.