ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ МОЛДОВЫ

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ, ИНФОРМАТИКИ И МИКРОЭЛЕКТРОНИКИ

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ОТЧЕТ

По лабораторной работе №1

по Програмированию для Windows

Тема: Простые приложения Windows

Выполнила: ст. гр. TI-155 Буянов Евгений

Проверил: Скроб С.

Кишинев 2017

1. **Цель лабораторной работы**

Изучение основ создания системных приложений.

1. **Теоретические понятия**

Windows – это графический интерфейс пользователя (Graphical User Interface, GUI), который иногда ещё называют «визуальный интерфейс» или «графическая оконная среда». Любая программа для Windows имеет окно – прямоугольную область на экране. Окно иденцифицируется заголовком. Оно получает от пользователя информацию в виде оконных «сообщений». У каждого окна, создаваемого программой, имеется соответствующая оконная процедура. Эта процедура является функцией, которая может находиться либо в самой программе, либо в динамически подключаемой библиотеке. Windows посылает сообщение окну путём вызова оконной процедуры, на основе этого сообщения окно совершает какие-то действия и затем возвращает управление Windows.Оконная процедура обрабатывает сообщения, поступающие окну. Когда программа для Windows начинает выполняться, Windows строит для программы очередь сообщений. В этой очереди храняться сообщения для любых типов окон, которые могли бы быть созданы программой.

Windows в настоящее время поддерживает свыше тысячи вызовов функций, которые можно использовать в приложениях. Каждая функция Windows имеет развёрнутое имя, написанное буквами как верхнего, так и нижнего регистров, например *CreateWindow*. Эта функция создаёт для нашей программы окно. Функция *IsClipboardAvailable* определяет, хранятся ли в буфере обмена данные специального формата.

Все основные функции Windows объявляются в заголовочных файлах. Главный заголовочный файл называется WINDOWS.H, и в этом файле содержится множество ссылок на другие заголовочные файлы.

Функции, необходимые для создания простого приложения:

* *LoadIcon –* загружает значок для использования в программе.
* *LoadCursor –* загружает курсор мыши для использования в программе.
* *GetStockObject –* получет графический объект (в этом случае для закрашивания фона окна используется кисть).
* *RegisterClassEx –* регистрирует класс окна для определённого окна программы.
* *CreateWindow –* создаёт окно на основе класса окна.
* *ShowWindow –* выводит окно на экран.
* *UpdateWindow –* заставляет окно перерисовать своё содержимое.
* *GetMessage –* получает сообщение из очереди сообщений.
* *TranslateMessage –* преобразует некоторые сообщения, полученные с помощью клавиатуры.
* *DispatchMessage –* отправляет сообщение оконной процедуре.
* *BeginPaint –* инициирует начало процесса рисования окна.
* *GetClientRect –* получает размер рабочей области окна.
* *DrawText –* выводит на экран строку текста.
* *EndPaint –* прекращает рисование окна.
* *PostQuitMessage –* вставляет сообщение «завершить» в очередь сообщений.
* *DefWindowProc –* выполняет обработку сообщений по умолчанию.

1. **Задание**

Написать программу для создания простого приложения. В центре рабочей области окна выводится сообщение. При изменении размеров окна сообщение остаётся в центре рабочей области.

* 1. **Листинг программы**

#include <windows.h>

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

int WINAPI WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, PSTR szCmdLine, int iCmdShow){

static char 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 = szAppName;

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

RegisterClassEx(&wndclass);

hwnd = CreateWindow(szAppName,

"Lab #1 Program",

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;

}

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

HDC hdc;

PAINTSTRUCT ps;

RECT rect;

switch (iMsg){

case WM\_PAINT:

hdc = BeginPaint(hwnd, &ps);

GetClientRect(hwnd, &rect);

DrawText(hdc, "- - - - - Laboratory work #1 of student from TI-155 group Zvercova Xenia - - - - -", -1, &rect, DT\_SINGLELINE | DT\_CENTER | DT\_VCENTER);

EndPaint(hwnd, &ps);

return 0;

case WM\_DESTROY:

PostQuitMessage(0);

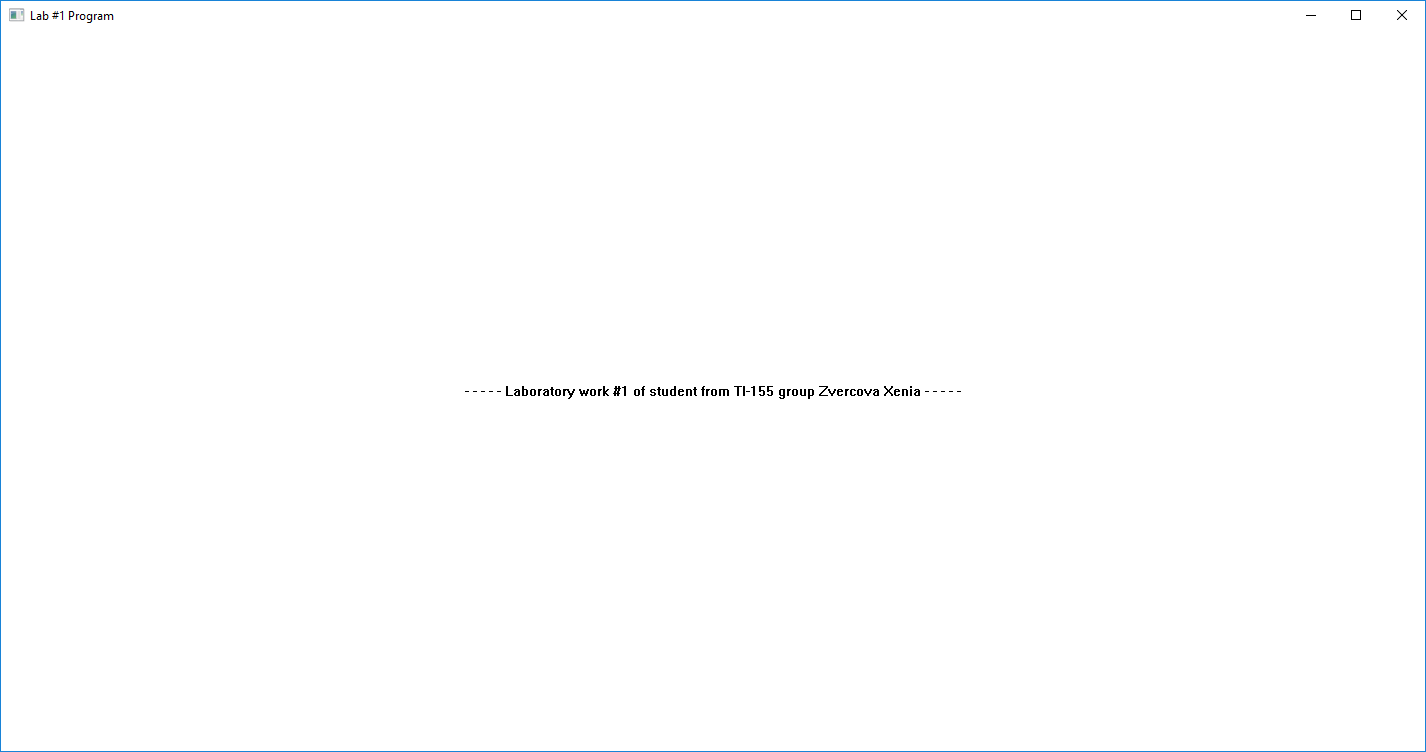
return 0;

}

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

}

* 1. **Результат работы прогрммы**



**Вывод:** В данной лабораторной работе были изучены основы создания простого приложения Windows,а также была реализована программа для создания простого приложения. Были изучены некоторые функции, необходимые для создания простого приложения. Windows – это графический интерфейс пользователя. А графика даёт лучшее восприятие действительного положения вещей на экране, визуально богатую среду для передачи информации и возможность WYSIWYG (What you see is what you get). Система Windows способствует тому, чтобы не нужно было тратить слишком много времени, чтобы составлять программы, поскольку все программы для Windows выглядят и воспринимаются одинаково.