МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ

КАФЕДРА ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ТА КОМП’ЮТЕРНОЇ ІНЖЕНЕРІЇ

Звіт з лабораторної роботи №1

з дисципліни

«Системне програмування для Windows»

|  |  |
| --- | --- |
| Перевірив:  ктн,доцент:  Миронюк Т.В | Виконав:  Студент 4 курсу  Групи КМ-175  Косенко А.В. |

Черкаси 2020

**Лабораторна робота №1**

**ЛАБОРАТОРНА РОБОТА № 1:** СТРУКТУРА WINDOWS-ДОДАТКІВ. ВІКОННА ПРОЦЕДУРА. ОБРОБКА ПОВІДОМЛЕНЬ

**Мета роботи:** ознайомлення зі структурою Windows - додатків, організацією взаємодії програм з операційною системою, отримання практичних навичок зі складання, написання і налагодження простих програм, що містять опис виду вікна додатка, віконну процедуру і оброблювання елементарних повідомлень від клавіатури, миші та інших пристроїв.

**Завдання на лабораторну роботу**

Написати з використанням функцій API програму, що створює віконний додаток Windows. Програма повинна містити реєстрацію класу вікна, процедуру створення вікна, віконну процедуру і цикл обробки повідомлень. Програма повинна виводити в клієнтській області вікна прізвище, ім'я, групу студента, який виконав завдання, і перемальовувати її при обробці повідомлення WM\_PAINT.

Варіант 2

Стиль класу: перемальовування вікна при зміні горизонтального розміру;

Тип завантажуваної іконки: IDI\_HAND,

Тип завантажуваного курсору: IDC\_WAIT,

Колір фону вікна: СВІТЛО СІРИЙ,

Ім'я класу вікна: ім'я студента (англійською мовою).

Стиль вікна:

а) вікно має кнопку мінімізації;

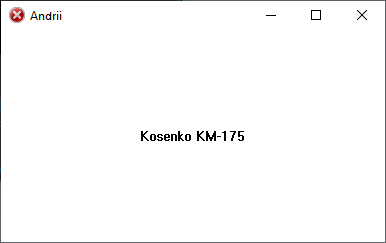
б) WS\_OVERLAPPEDWINDOW.

Позиція вікна: в центрі екрану шириною 400 висотою 250 точок.

**Лістинг програми**

|  |
| --- |
| #include <windows.h>  #include <tchar.h>  const TCHAR szWindowClassName[] = \_T("Kosenko");  const TCHAR szTitle[] = \_T("Andrii");  const TCHAR text[] = \_T("Kosenko KM-175");  // Глобальні змінні:  HINSTANCE hInst; // Вказівник додатку  // Попередній опис функції  ATOM MyRegisterClass(HINSTANCE hInstance);  BOOL InitInstance(HINSTANCE, int);  LRESULT CALLBACK WndProc(HWND, UINT, WPARAM, LPARAM);  // Головна програма  int APIENTRY \_tWinMain(HINSTANCE hInstance,  HINSTANCE hPrevInstance,  LPTSTR lpCmdLine,  int nCmdShow)  {  MSG msg;  // Реєстрація класу вікна  MyRegisterClass(hInstance);  // Створення вікна додатку  if (!InitInstance(hInstance, nCmdShow))  {  return FALSE;  }  // Цикл обробки повідомлень  while (GetMessage(&msg, NULL, 0, 0))  {  TranslateMessage(&msg);  DispatchMessage(&msg);  }  return msg.wParam;  }  // FUNCTION: MyRegisterClass()  // Реєструє клас вікна  ATOM MyRegisterClass(HINSTANCE hInstance)  {  WNDCLASSEX wcex;  wcex.cbSize = sizeof(WNDCLASSEX);  wcex.style = CS\_HREDRAW | CS\_VREDRAW | CS\_NOCLOSE;// стиль вікна  wcex.lpfnWndProc = WndProc; // віконна процедура wcex.lpfnWndProc = (WNDPROC)WndProc;  wcex.cbClsExtra = 0;  wcex.cbWndExtra = 0;  wcex.hInstance = hInstance; // вказівник додатку  wcex.hIcon = LoadIcon(NULL, MAKEINTRESOURCE(IDI\_HAND)); // визначення іконки  wcex.hCursor = LoadCursor(NULL, IDC\_WAIT); // визначення курсора  wcex.hbrBackground = GetSysColorBrush(RGB(128, 128, 128));// встановлення фону  wcex.lpszMenuName = NULL; // визначення меню  wcex.lpszClassName = szWindowClassName; // ім’я класу  wcex.hIconSm = NULL;  return RegisterClassEx(&wcex); // реєстрація класу вікна  }  // FUNCTION: InitInstance(HANDLE, int)  // Створює вікно додатку і зберігає вказівник додатку в змінній hInst  BOOL InitInstance(HINSTANCE hInstance, int nCmdShow)  {  HWND hWnd;  hInst = hInstance; // зберігає вказівник додатку в змінній hInst  hWnd = CreateWindow(szWindowClassName, // ім’я класу вікна  szTitle, //ім’я додатку  WS\_OVERLAPPEDWINDOW, // стиль вікна  CW\_USEDEFAULT, // положення по Х  CW\_USEDEFAULT, // положення по Y  400, // розмір по Х  250, // розмір по Y  NULL, // дескриптор батьківського вікна  NULL, // дескриптор меню вікна(дескриптор меню вікна)  hInstance, // вказівник додатку  NULL); // параметри створення.  if (!hWnd) // Якщо вікно не створилось, функція повертає FALSE  {  return FALSE;  }  ShowWindow(hWnd, nCmdShow); // Показати вікно  UpdateWindow(hWnd); // Обновити вікно  return TRUE; //Успішне завершення функції  }  // FUNCTION: WndProc(HWND, unsigned, WORD, LONG)  // Віконна процедура. Приймає і оброблює всі повідомлення, які надходять в додаток  LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam)  {  PAINTSTRUCT ps;  HDC hdc;  RECT rt;  switch (message)  {  case WM\_CREATE: // Повідомлення приходить при створенні вікна  break;  case WM\_PAINT: // Перемалювати вікно  hdc = BeginPaint(hWnd, &ps); // Почати графічне виведення  GetClientRect(hWnd, &rt); // Область вікна для малювання  DrawText(hdc, text, -1, &rt, DT\_SINGLELINE | DT\_CENTER | DT\_VCENTER);  EndPaint(hWnd, &ps); // Закінчити графічне виведення  break;  case WM\_DESTROY: // Завершення роботи  PostQuitMessage(0);  break;  default:  // Обробка повідомлень, які не оброблені користувачем  return DefWindowProc(hWnd, message, wParam, lParam);  }  return 0;  } |

**Результат виконання лабораторної роботи**



**Висновки:** було ознайомлено зі структурою Windows - додатків, організацією взаємодії програм з операційною системою, отримання практичних навичок зі складання, написання і налагодження простих програм, що містять опис виду вікна додатка, віконну процедуру і оброблювання елементарних повідомлень від клавіатури, миші та інших пристроїв.