Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«Комсомольский-на-Амуре государственный университет»

Факультет компьютерных технологий

Кафедра «МОП ЭВМ»

ЛАБОРАТОРНАЯ РАБОТА 2

по дисциплине «Операционные системы и оболочки»

## Обработка сообщений от элементов управления экранных форм

Студент группы 6ИСб-1 А.М. Закусило

Преподаватель В.А Тихомиров

2018

**Цель работы**: Разместить элементы формы и настроить обработчики событий на эти элементы.

**Задание**: Создать программный модуль по технологии Win API, содержащий три кнопки, графическое окно, раскрывающийся список. В списке два режима: ручной и автомат. При нажатии на первую кнопку – в окне рисуется круг, вторую – прямоугольник, третью – линия случайной толщины, цвета и заливки. А если включен автомат, то объекты рисуются непрерывно в случайном месте окна.

**Выполнение:**

Текст программы приведен в листинге 1.

Листинг 1 – Текст файла Prog2.cpp

|  |
| --- |
| #include <windows.h>  #include "resource1.h"  #include <stdlib.h>  #include <stdio.h>  // Три кнопки, графическое окно, раскрывающийся список.  // В списке два режима : ручной и автомат. При нажатии на первую кнопку – в окне рисуется круг,  // вторую – прямоугольник, третью – линия случайной толщины, цвета и заливки.  // А если включен автомат, то объекты рисуются непрерывно в случайном месте окна.  HWND hCombo1; // окно листбокс  HWND hWnd; // главное окно  HANDLE h; // второй поток  HINSTANCE ghInstance; // Переменная для хранения хендела процесса  // Описание используемой оконной процедуры  BOOL CALLBACK PviewDlgProc(HWND hWnd, UINT wMsg, WPARAM wParam, LPARAM lParam);  DWORD WINAPI thread2(LPVOID t)  {  InvalidateRect(hWnd, NULL, TRUE);  UpdateWindow(hWnd);  int RAND;  //генерируем элементы строки  while (true)  {  RAND = rand() % 3;  switch (RAND)  {  case 0:  {  HPEN pen;  HDC hdc = GetDC(hWnd);  int x = rand() % 800;  int y = rand() % 450;  int z = rand() % 255;  int width = rand() % 10;  pen = CreatePen(PS\_SOLID, width, RGB(x, y, z));  SelectObject(hdc, pen);  MoveToEx(hdc, x, y, 0); //перемещение пера  LineTo(hdc, 600 - y, 230 - x/2); //рисовать линию  DeleteObject(pen); // удалить карандаш  ReleaseDC(hWnd, hdc); // освободить девайс контекста  }  break;  case 1:  {  HDC hdc = GetDC(hWnd);  double x = rand() % 10 + 1;  x = x / 10;  int y = rand() % 300;  int z = rand() % 5;  Rectangle(hdc, 260 \* x - (y\*z), 160 \* x - (y\*z), 575 \* x - (y\*z), 300 \* x - (y\*z));  ReleaseDC(hWnd, hdc);  }  break;  case 2:  {  HDC hdc = GetDC(hWnd);  double x = rand() % 10;  x = x / 5;  Ellipse(hdc, 300\* x, 340 \* x, 530 \* x, 120\* x);  ReleaseDC(hWnd, hdc);  }  break;  default:  break;  }  Sleep(1000);  }  return 0;  }  // Главное приложение программы  int WINAPI WinMain(HINSTANCE hInstance,  HINSTANCE hPrevInstance,  LPSTR lpCmdLine,  int nCmdShow)  {  MSG msg;  ghInstance = hInstance;  // Создание диалогового окна  hWnd = CreateDialogParam(hInstance,  MAKEINTRESOURCE(PVIEW\_DLG),  NULL,  (DLGPROC)PviewDlgProc,  (LONG)0);  // Стандартный цикл обработки сообщений приложения  while (GetMessage(&msg, NULL, 0, 0))  if (!IsDialogMessage(hWnd, &msg))  {  TranslateMessage(&msg);  DispatchMessage(&msg);  }  DestroyWindow(hWnd);  return 0;  }  //Процедура обработки сообщений диалогового окна  BOOL CALLBACK PviewDlgProc(HWND hWnd,  UINT wMsg,  WPARAM wParam,  LPARAM lParam)  {  switch (wMsg)  {  case WM\_INITDIALOG:  {  hCombo1 = GetDlgItem(hWnd, IDC\_COMBO1);  SendMessage(hCombo1, CB\_ADDSTRING, 0, (DWORD)TEXT(  "Ручной режим"));  SendMessage(hCombo1, CB\_ADDSTRING, 0, (DWORD)TEXT(  "Автомат"));  // Выбор эл по умолчанию  SendMessage(hCombo1, CB\_SETCURSEL, 0, 0);  }  break;  case WM\_CLOSE:  {  PostQuitMessage(0);  }  break;  // Сообщение от элементов диалогового окна  case WM\_COMMAND:  {  switch (LOWORD(wParam))  {  case IDC\_CIRCLE:  {  HDC hdc = GetDC(hWnd);  InvalidateRect(hWnd, NULL, TRUE);  UpdateWindow(hWnd);  Ellipse(hdc, 300, 340, 530, 120);  ReleaseDC(hWnd, hdc);  // char szText[256]="hello";  // GetDlgItemText(hWnd, IDC\_EDIT1, (LPSTR)szText, 255);  // SetWindowText(GetDlgItem(hWnd, IDC\_EDIT1), "");  // MessageBox(hWnd, szText, "Принят ТЕКСТ:", MB\_OK);  }  break;  case IDC\_BOX:  {  HDC hdc = GetDC(hWnd);  InvalidateRect(hWnd, NULL, TRUE);  UpdateWindow(hWnd);  Rectangle(hdc, 260, 160, 575, 300);  ReleaseDC(hWnd, hdc);  }  break;  case IDC\_LINES:  {  InvalidateRect(hWnd, NULL, TRUE);  UpdateWindow(hWnd);  HPEN pen;  HDC hdc = GetDC(hWnd);  int width = rand() % 10;  int x = rand() % 255;  int y = rand() % 255;  int z = rand() % 255;  pen = CreatePen(PS\_SOLID, width, RGB(x, y, z));  SelectObject(hdc, pen);  MoveToEx(hdc, 225, 230, 0); //перемещение пера  LineTo(hdc, 600, 230); //рисовать линию  DeleteObject(pen); // удалить карандаш  ReleaseDC(hWnd, hdc); // освободить девайс контекста  }  break;  case IDC\_COMBO1:  {  if (HIWORD(wParam) == CBN\_SELCHANGE)  {  char str[8];  char str2[] = "Автомат";  GetDlgItemText(hWnd, IDC\_COMBO1, str, 8);  if (strcmp(str, str2) == 0)  {  h = CreateThread(NULL, 0, thread2, NULL, 0, NULL);  }  else  {  InvalidateRect(hWnd, NULL, TRUE);  UpdateWindow(hWnd);  TerminateThread(h, 1);  }  }  }  break;  default:  return FALSE;  }  }  break;  default:  return FALSE;  }  return TRUE;  } |

Результат работы программы приведены на рисунках 1-2.

Программа может работать в двух режимах. Ручной режим работы. (рисунок 1):

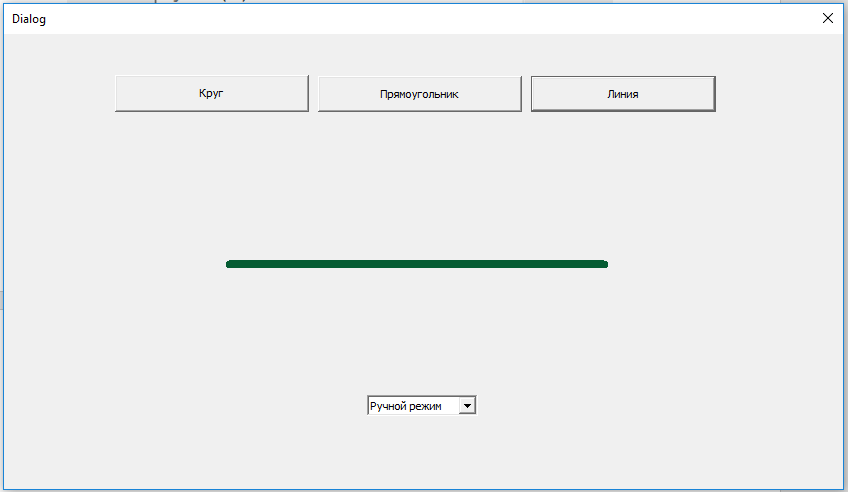


Рисунок 1 – Ручной режим работы

Программа может работать в автоматическом режиме (рисунок 2):

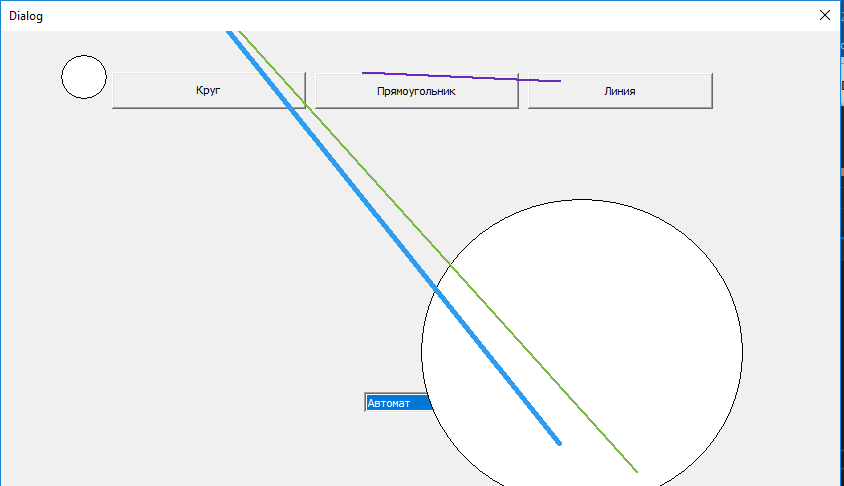


Рисунок 2 – Автоматический режим работы

**Список использованных источников**

1. Саймон, Р. Windows 2000 API Энцеклопедия программиста / Р. Саймон. - М: DiaSoft, 2002