**Практическая работа №7**

**«Обработка одномерных массивов с использованием DLL-библиотек»**

В данной практической работе нам необходимо написать программу на С++ с использованием DLL-библиотеки которая создает массив, с учётом условия

(число делится на 2 и не делится на 3) создает результирующий массив, выводит его и записывает его в базу данных Access (вместе с исходным).

Ниже представлен код DLL-библиотеки:

**.h**

#pragma once

using namespace System;

namespace ClassLibrary1 {

public ref class ClassLib

{

public:

static void genmas(int\* mas, int len);

static int setrezmas(int\* mas, int\* rezmas, int len);

};

}

**.cpp**

#include "stdafx.h"

#include "ClassLibrary1.h"

#include <random>

#include <time.h>

namespace ClassLibrary1 {

void ClassLib::genmas(int\* mas, int len) {

srand(time(NULL));

for (int i = 0; i < len; i++)

mas[i] = -20 + rand() % 45;

}

int ClassLib::setrezmas(int\* mas, int\* rezmas, int len) {

int j = 0;

for (int i = 0; i < len; i++) {

if ((mas[i] % 2 == 0) && (mas[i] % 3 != 0)) {

rezmas[j] = mas[i];

j++;

}

}

return j;

}

}

Ниже представлен код основной формы:

#pragma endregion

private: System::Void button1\_Click(System::Object^ sender, System::EventArgs^ e) {

ADOX::Catalog ^ Catalog = gcnew ADOX::CatalogClass();

try

{

Catalog->Create("Provider=Microsoft.Jet." + "OLEDB.4.0;Data Source=new\_BD.mdb");

MessageBox::Show("База данных new\_BD.mdb успешно создана");

}

catch (System::Runtime::InteropServices::COMException ^ Ситуация)

{

MessageBox::Show(Ситуация->Message);

}

finally

{

Catalog = nullptr;

}

}

private: System::Void button2\_Click(System::Object^ sender, System::EventArgs^ e) {

// ЗАПИСЬ СТРУКТУРЫ ТАБЛИЦЫ В ПУСТУЮ БД:

// Создание экземпляра объекта Connection с указанием строки

// подключения:

auto Connection = gcnew OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=new\_BD.mdb");

// Открытие подключения:

Connection->Open();

// Создание экземпляра объекта класса Command

// с заданием SQL-запроса:

auto Command = gcnew OleDbCommand("CREATE TABLE [Massives] ([Mas] string, [Rezmas] string)", Connection);

try // Выполнение команды SQL:

{

Command->ExecuteNonQuery();

MessageBox::Show("Структура таблиц записана в пустую БД");

}

catch (Exception ^ Situation)

{

MessageBox::Show(Situation->Message);

}

Connection->Close();

delete Command;

delete Connection;

}

private: System::Void button4\_Click(System::Object^ sender, System::EventArgs^ e) {

int len = Convert::ToInt32(textBox1->Text);

int\* mas = new int[len];

int\* rezmas = new int[len];

ClassLibrary1::ClassLib::genmas(mas, len);

int lenrez = ClassLibrary1::ClassLib::setrezmas(mas, rezmas, len);

int j = 0;

for (int i = 0; i < len; i++) {

if (j >= lenrez)

rezmas[j] = 0;

dataGridView1->Rows->Add();

dataGridView1->Rows[i]->Cells[0]->Value = mas[i].ToString();

dataGridView1->Rows[i]->Cells[1]->Value = rezmas[i].ToString();

auto Connection = gcnew OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=new\_BD.mdb");

auto Command = gcnew OleDbCommand("INSERT INTO Massives (Mas, Rezmas) VALUES (" + mas[i] + "," + rezmas[j] +")");

Connection->Open();

Command->Connection = Connection;

Command->ExecuteNonQuery();

Connection->Close();

delete Connection, Command;

j++;

}

MessageBox::Show("В таблицы добавлены записи");

}

Ниже представлены скриншоты работы программы:

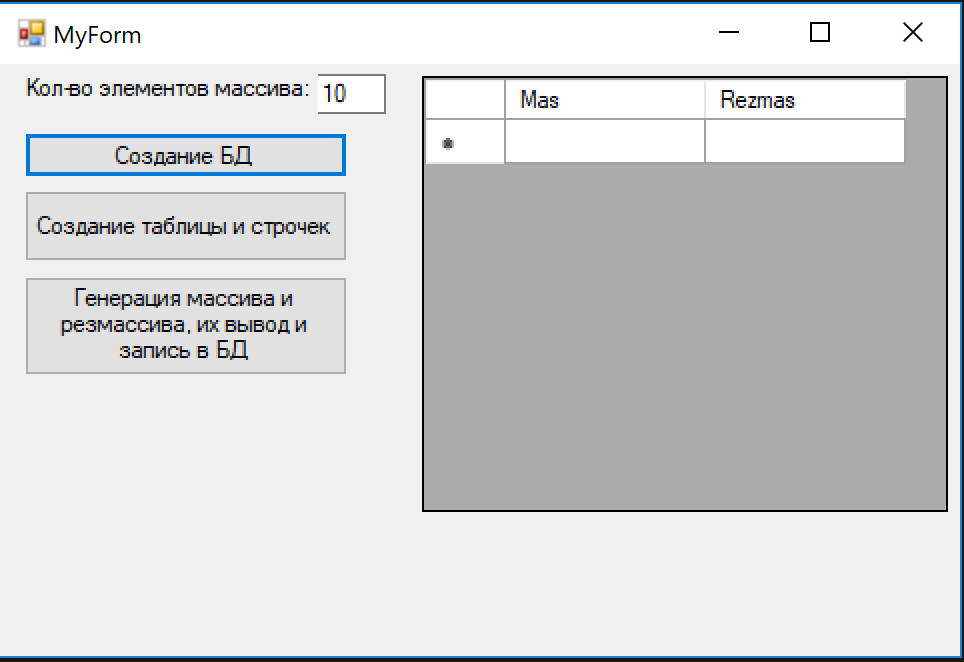


Рисунок 1 - Основная форма

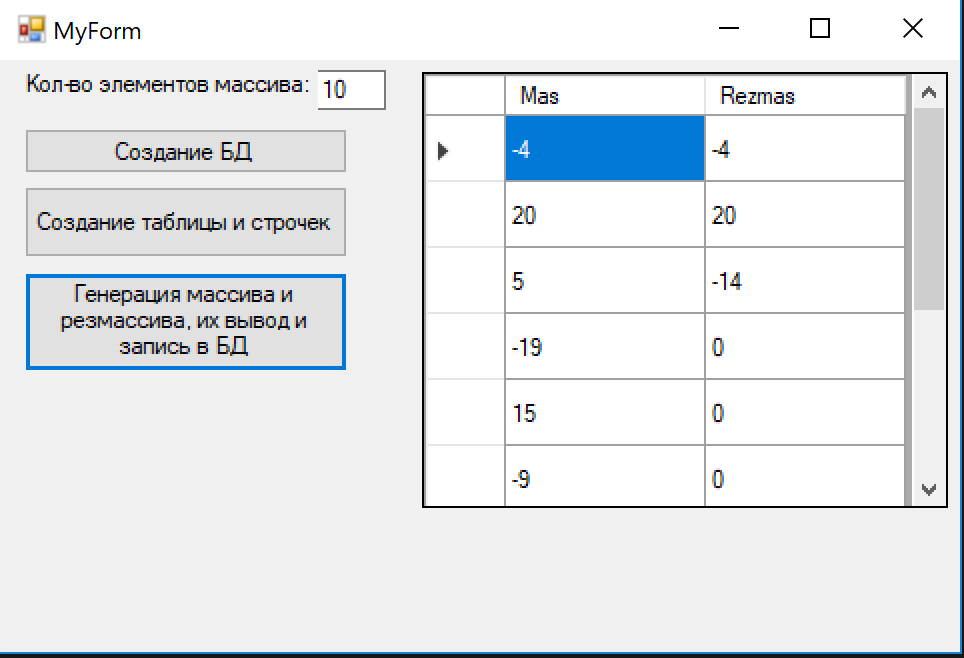


Рисунок 2 - Форма после отработки 3-х кнопок

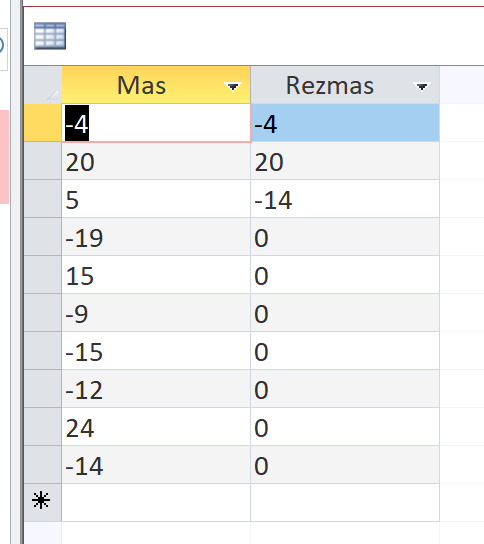


Рисунок 3 - Записанные данные в БД