Лабораторная работа № 8  
на тему  
**«Работа с одномерными массивами»**

# СОДЕРЖАНИЕ

СОДЕРЖАНИЕ 2

1 Задание 3

2 Основная часть 4

2.1 Блок-схемы 4

2.2 Исходный код программы 7

2.2.1 Библиотека 7

2.2.2 Основная программа 8

3 Заключение 10

# 1 Задание

Дан динамический целочисленный массив. Элементы массива могут принимать целые значения от 0 до 1000 включительно. Опишите алгоритм, который подсчитывает количество пар соседних элементов, разница между которыми не кратна 7. Гарантируется, что в массиве есть хотя бы один элемент, кратный 7.

Разработайте функцию, которая формирует новый массив из тех элементов исходного массива, значения которых делится на найденное число без остатка и выведите исходный и результирующий массивы в Microsoft Access.

Для выполнения данного задания необходимо дополнить DLL-библиотеку из прошлой работы следующими функциями:

1. solve2(int\* array, int\* new\_array, int length, int& new\_length, int number), которая формирует новый массив из тех элементов исходного массива, значения которых делится на число, полученное в результате выполнения первой части задания с использованием ранее созданной функции solve(…), без остатка;
2. create\_db(), которая создаёт базу данных для записи полученных результатов;
3. configure\_db(), которая создает таблицу с необходимым таблицами и разметкой:
4. write\_to\_db(int\* array, int\* new\_array, int length, int new\_length), которая записывает исходный и результирующий массивы в базу данных.

Также необходимо разработать событийную функцию button\_create\_click(..), обрабатывающую нажатие кнопки «СОЗДАТЬ ТАБЛИЦУ», которая создаёт базу данных в рабочей директории и событийную функцию button\_fill\_click(..), которая выполняет расчёты массивов, выводит их в графические элементы и записывает в базу данных.

# 2 Основная часть

## 2.1 Блок-схемы

Блок-схемы, реализованные в тексте программы, указаны ниже.

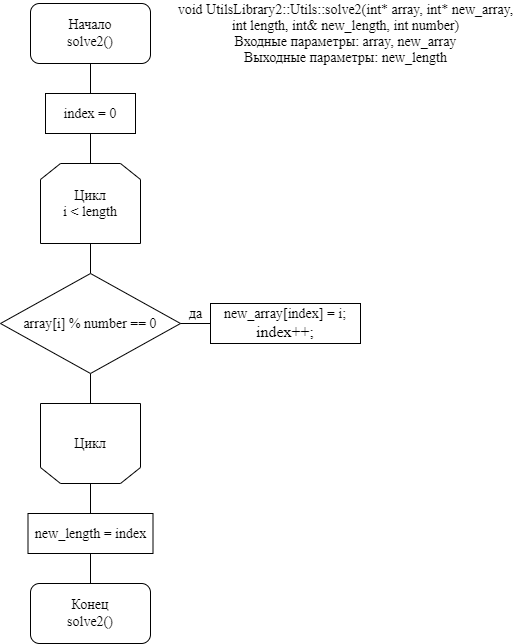


Рисунок 1 — Функция solve2()

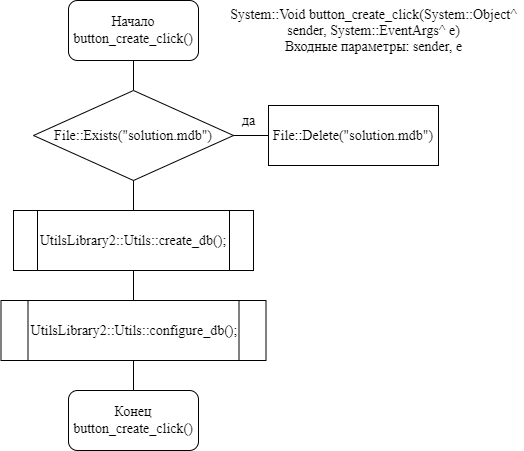


Рисунок 2 — Функция button\_ create\_click()

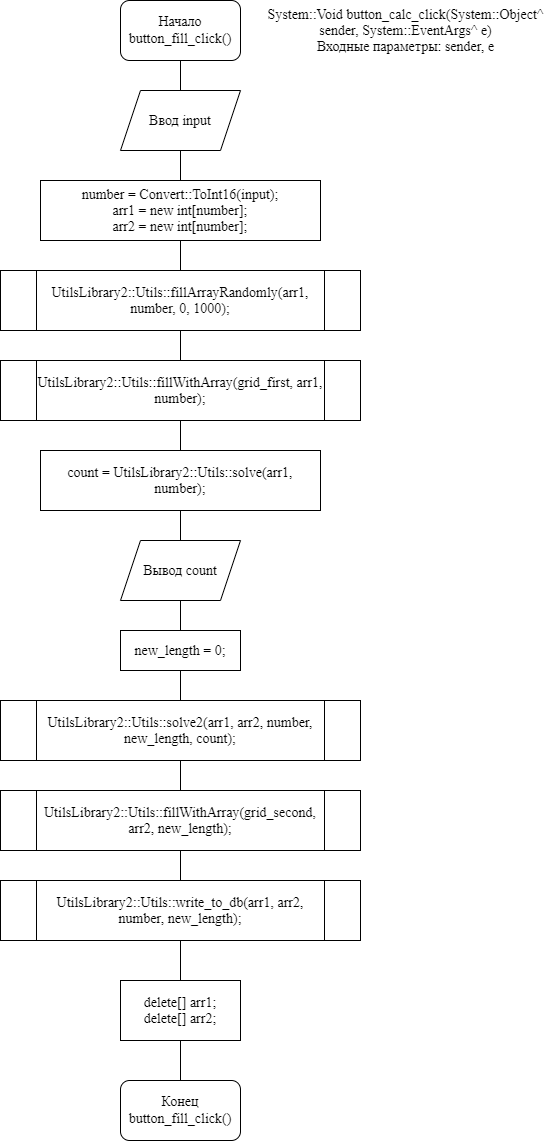


Рисунок 3 — Функция button\_ fill\_click()

## 2.2 Исходный код программы

### 2.2.1 Библиотека

void UtilsLibrary2::Utils::solve2(int\* array, int\* new\_array, int length, int& new\_length, int number)

{

int index = 0;

for (size\_t i = 0; i < length; i++)

{

if (array[i] % number == 0)

{

new\_array[index] = i;

index++;

}

}

new\_length = index;

}

void UtilsLibrary2::Utils::create\_db()

{

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

try

{

catalog->Create("Provider = Microsoft.Jet.OLEDB.4.0; Data Source = solution.mdb");

MessageBox::Show("Файл базы данных успешно создан", "Инфо", MessageBoxButtons::OK, MessageBoxIcon::Information);

}

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

{

MessageBox::Show(exception->Message, "Ошибка", MessageBoxButtons::OK, MessageBoxIcon::Information);

}

finally

{

catalog = nullptr;

}

}

void UtilsLibrary2::Utils::configure\_db()

{

auto p = gcnew OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source = solution.mdb");

p->Open();

auto c = gcnew OleDbCommand("CREATE TABLE [Результат]([Индекс] counter, [Исходный массив] char(200)," +

"[Результирующий массив] char(200))", p);

try

{

c->ExecuteNonQuery();

MessageBox::Show("Структура базы данных успешно сформирована", "Инфо", MessageBoxButtons::OK, MessageBoxIcon::Information);

}

catch (OleDbException^ exception)

{

MessageBox::Show(exception->Message, "Ошибка", MessageBoxButtons::OK, MessageBoxIcon::Information);

}

p->Close();

}

void UtilsLibrary2::Utils::write\_to\_db(int\* array, int\* new\_array, int length, int new\_length)

{

for (size\_t i = 0; i < length; i++)

{

auto p = gcnew OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source = solution.mdb");

p->Open();

if (i < new\_length)

{

auto c = gcnew OleDbCommand("INSERT INTO[Результат](" +

"[Исходный массив], [Результирующий массив]) VALUES('" + array[i] + "','" + new\_array[i] + " ')");

c->Connection = p;

c->ExecuteNonQuery();

}

else

{

auto c = gcnew OleDbCommand("INSERT INTO[Результат](" +

"[Исходный массив], [Результирующий массив]) VALUES('" + array[i] + "','')");

c->Connection = p;

c->ExecuteNonQuery();

}

}

}

### 2.2.2 Основная программа

System::Void button\_create\_click(System::Object^ sender, System::EventArgs^ e)

{

if (System::IO::File::Exists("solution.mdb"))

{

// Следует создать новый файл базы данных,

// чтобы не мешать результаты с прошлыми

System::IO::File::Delete("solution.mdb");

}

UtilsLibrary2::Utils::create\_db();

UtilsLibrary2::Utils::configure\_db();

}

System::Void button\_fill\_click(System::Object^ sender, System::EventArgs^ e)

{

button\_fill->Enabled = false;

String^ input = Interaction::InputBox("Введите количество элементов массива", "Ввод", "10", -1, -1);

int number = Convert::ToInt16(input);

int\* arr1 = new int[number];

int\* arr2 = new int[number];

UtilsLibrary2::Utils::fillArrayRandomly(arr1, number, 0, 1000);

UtilsLibrary2::Utils::fillWithArray(grid\_first, arr1, number);

size\_t count = UtilsLibrary2::Utils::solve(arr1, number);

MessageBox::Show("Количество пар соседних элементов, разница\nмежду которыми НЕ кратна 7: " +

Convert::ToString(count), "Вывод", MessageBoxButtons::OK,

MessageBoxIcon::Information);

int new\_length = 0;

UtilsLibrary2::Utils::solve2(arr1, arr2, number, new\_length, count);

UtilsLibrary2::Utils::fillWithArray(grid\_second, arr2, new\_length);

UtilsLibrary2::Utils::write\_to\_db(arr1, arr2, number, new\_length);

delete[] arr1;

delete[] arr2;

button\_fill->Enabled = true;

}

# 3 Заключение

Техническое задание были успешно реализовано на языке программирования Visual C++. Программа работает корректно, полученные результаты, представленные на рисунках ниже, верны.

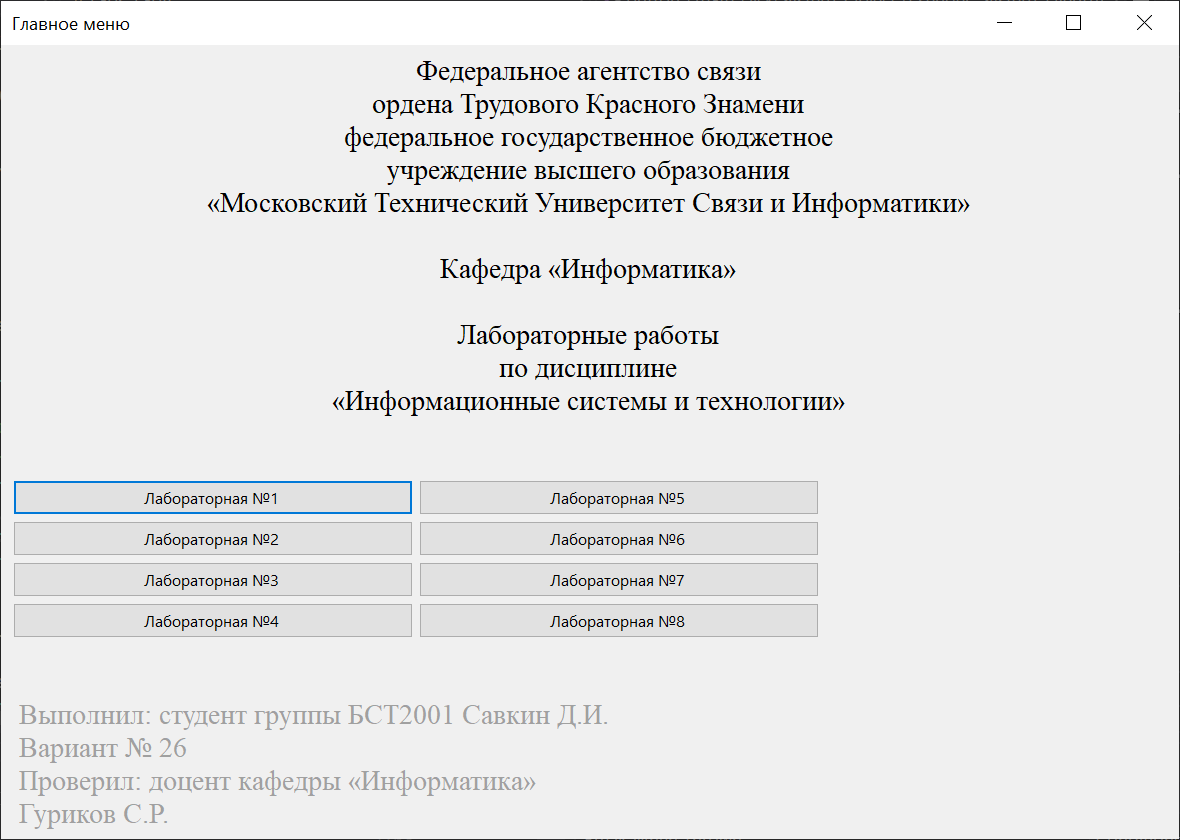


Рисунок 4 — Главное меню программы

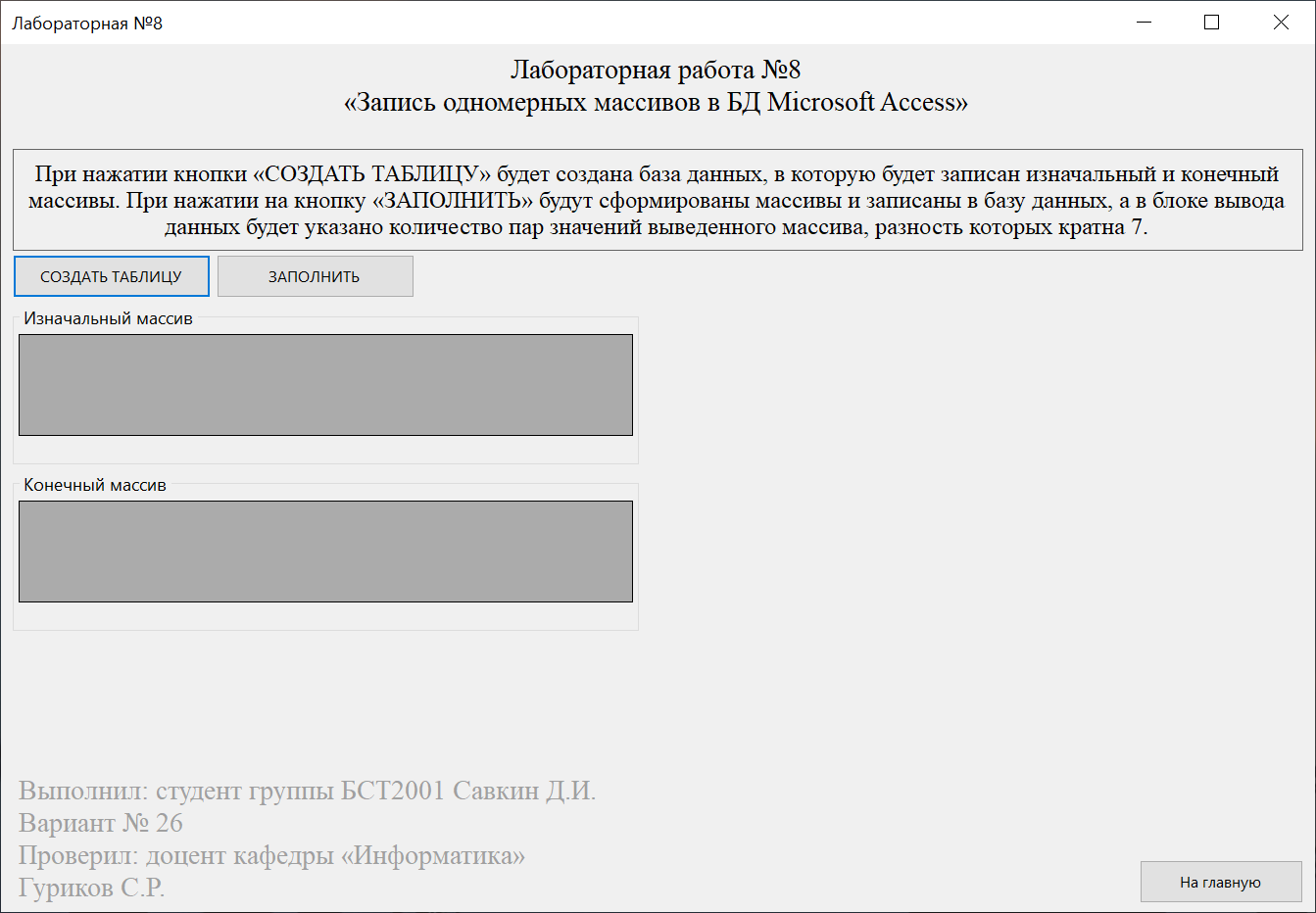


Рисунок 5 — Форма лабораторной работы

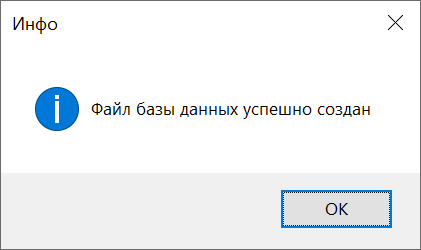


Рисунок 6 — Отчёт об успешном создании файла  
базы данных

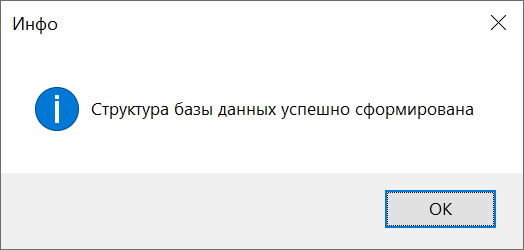


Рисунок 7 — Отчёт об успешном создании структуры файла  
базы данных

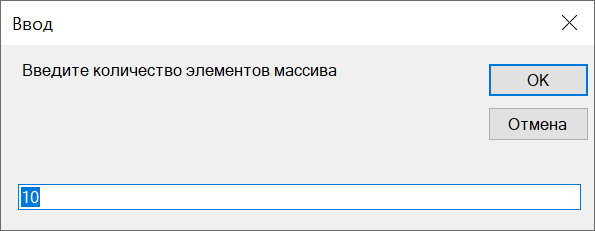


Рисунок 8 — Организация ввода размера массива

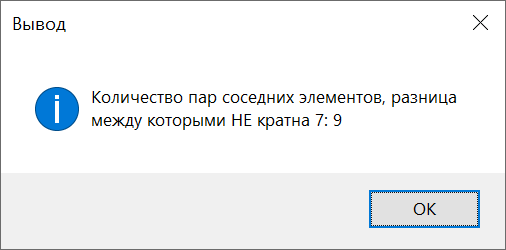


Рисунок 9 — Организация вывода результатов

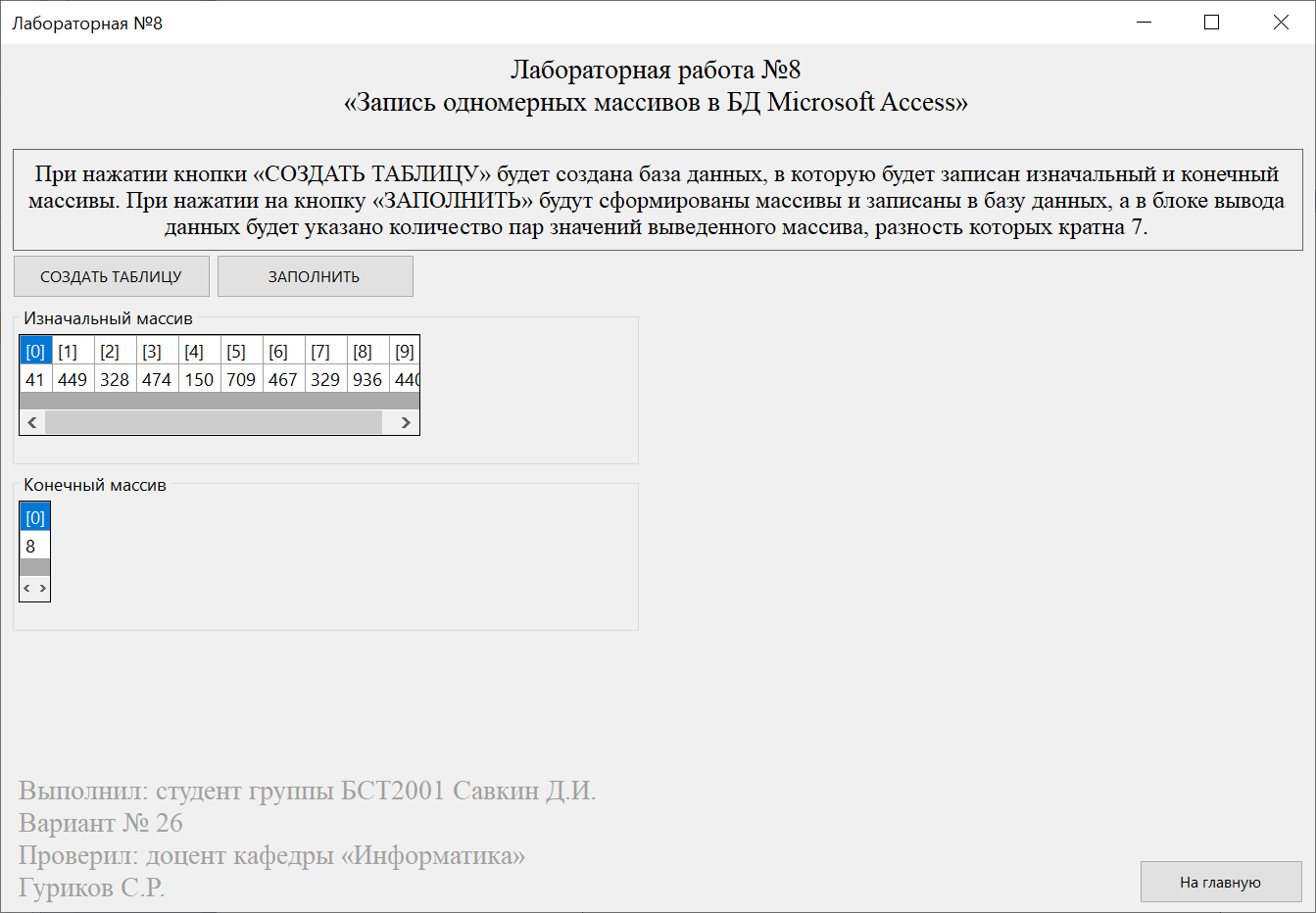


Рисунок 10 — Вывод результатов

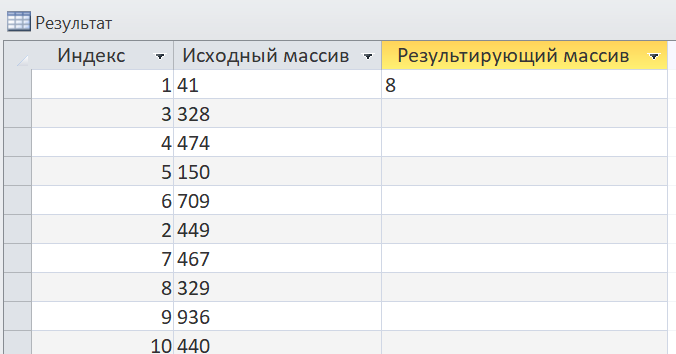


Рисунок 10 — База данных