Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики

и радиоэлектроники»

Специальность «Программная инженерия»

Кафедра инженерной психологии и эргономики

Учебная дисциплина «Основы алгоритмизации и программирования»

ОТЧЕТ

по лабораторной работе No4

«Массивы»

Вариант 7

Подготовил: Гавлев Н.В.

Проверил: Усенко Ф.В.

Минск 2024

**Цель работы:** сформировать навыки и умения обработки структурированных типов данных, организованных в виде некоторой линейной последовательности, а также организованных в виде матрицы.

Задание 1. Преобразовать массив К размером n таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях.

Задание 2. Дан массив размерностью n\*m. Поделить все элементы каждой строки на наибольший по модулю элемент этой строки.

1. Одномерные массивы:

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "RU");

const int n = 10;

int k[n];

int newk [n];

int oddindex = 0;

int evenindex = n / 2;

cout << "Если хотите ввести свои значения в массив, введите 'input' или массив будет заполнен числами от 1 до " << n << ": ";

string answer;

cin >> answer;

if (answer == "input")

{

cout << "Введите элементы массива: ";

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

{

cin >> k[i];

}

}

else

{

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

{

k[i] = i + 1;

}

}

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

{

if ((i + 1) % 2 == 1)

{

newk[oddindex] = k[i];

oddindex += 1;

}

else

{

newk[evenindex] = k[i];

evenindex += 1;

}

}

cout << "Преобразованный массив: ";

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

{

cout << newk[i] << " ";

}

}

1. Двумерные массивы:

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "RU");

const int lines = 4, columns = 4;

float new\_my\_list[lines][columns];

float max\_value = 0;

int my\_list[lines][columns]

{

{ 1, 2, 3, -4 },

{ 5, 6, 7, 8 },

{ 9, 10, 11, 12 },

{ 13, 14, 15, 16 },

};

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

{

for (int j = 0; j < columns; j++)

{

if (fabs(my\_list[i][j]) > fabs(max\_value))

{

max\_value = my\_list[i][j];

}

}

for (int j = 0; j < columns; j++)

{

new\_my\_list[i][j] = my\_list[i][j] / max\_value;

}

max\_value = 0;

}

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

{

for (int j = 0; j < columns; j++)

{

cout << "Значение массива c номером строки " << i + 1 << " и номером столбца " << j + 1 << " поделенное на максимальное по модулю значение в строке это ";

cout << new\_my\_list[i][j] << endl;

}

}

}

На рисунках 1-3 показаны скриншоты работающей программы.

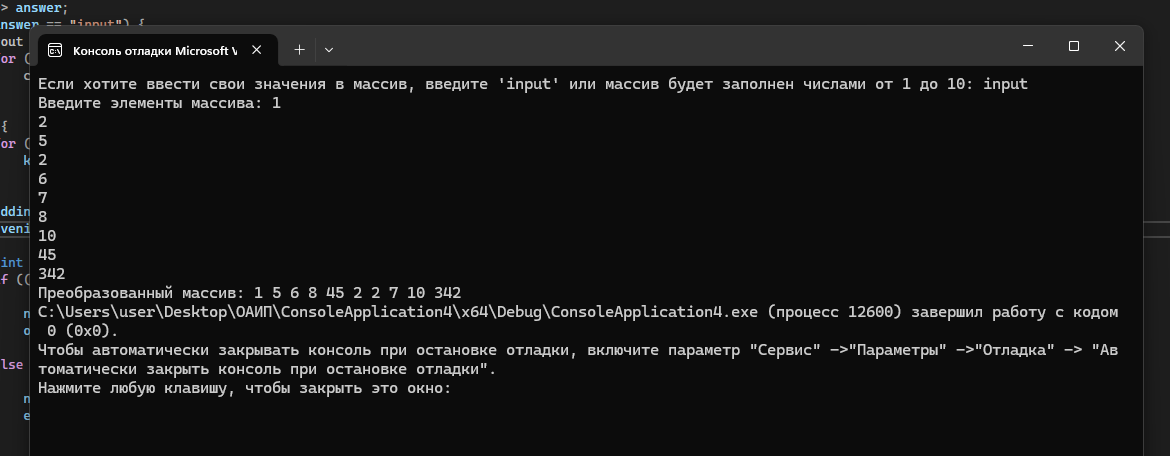


Рисунок 1 – Работа программы в задании с одномерными массивами при вводе слова input

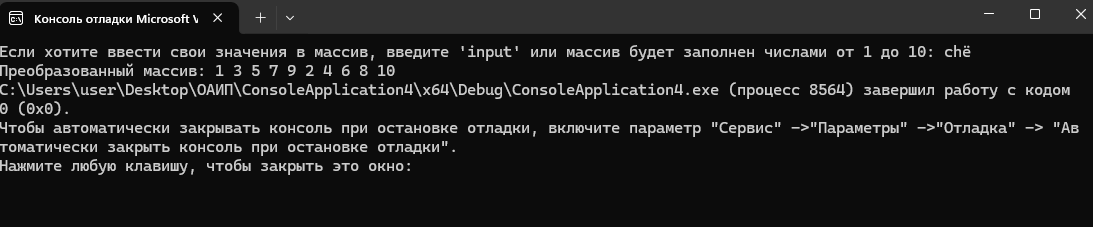


Рисунок 2 – Работа программы в задании с одномерными массивами при вводе случайных символов

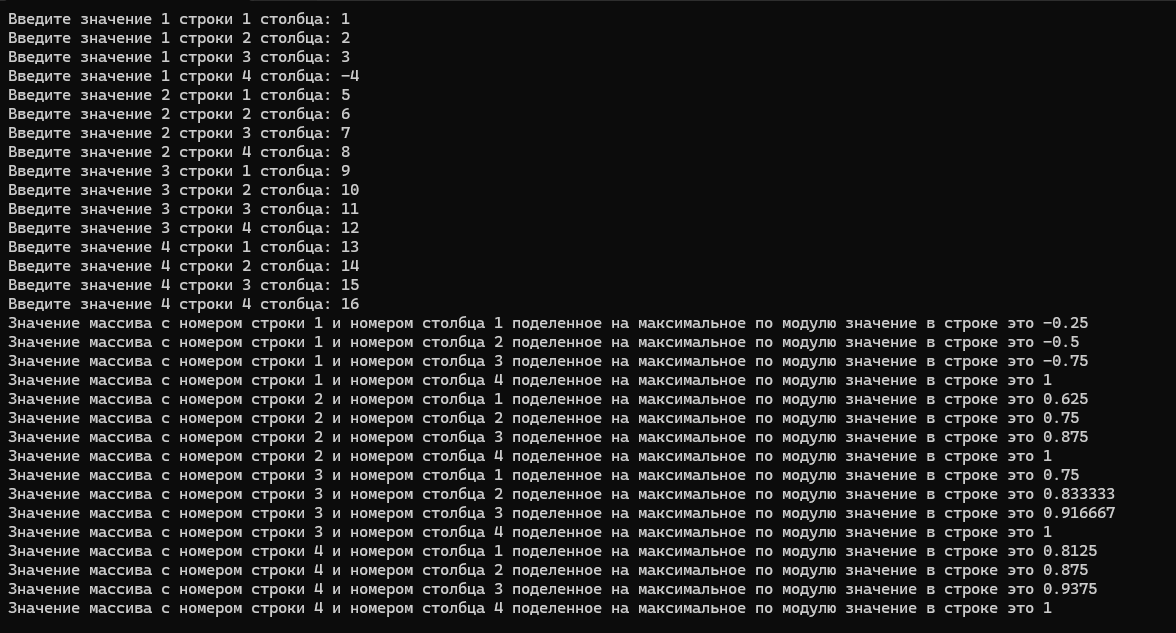
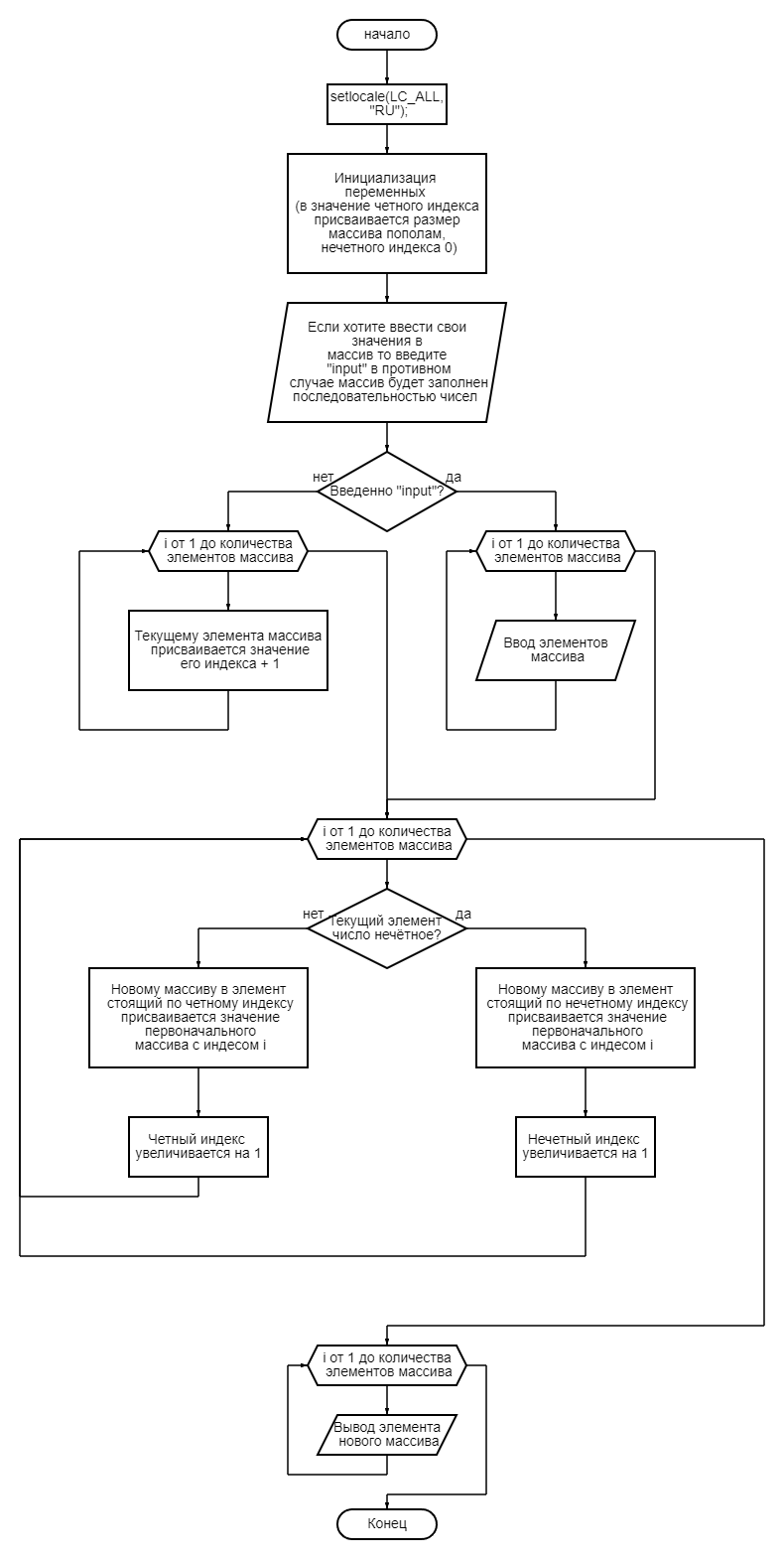
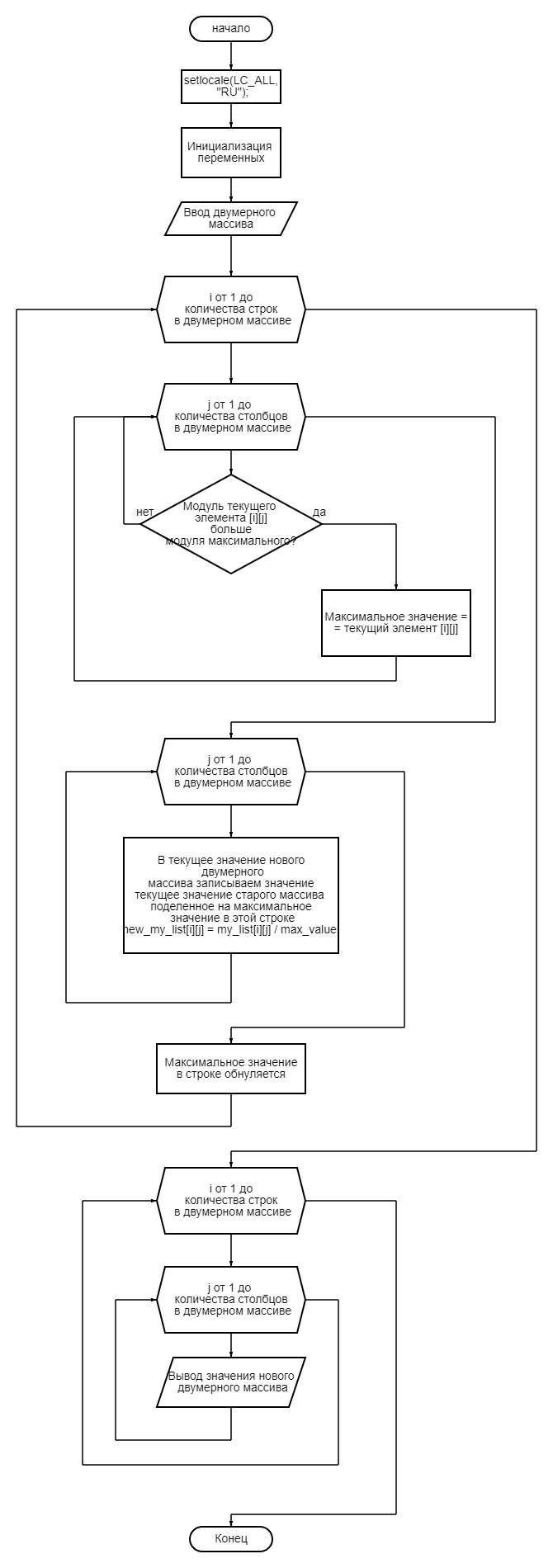


Рисунок 3 – Работа программы в задании с двумерными массивами

Далее приведены блок схемы





**Вывод:** В ходе выполнения лабораторной работы создано 2 приложения. В программе с одномерными массивами реализован ввод своих значений в массив или заполнение их по заданному изначально паттерну. После этого они сортируются по положению в массиве, нечетные становятся в начале, четные с середины по конец. При этом четность/нечетность номера элемента массива начинается не с 0, а с 1, так как так проще воспринимать массив.

В программе с двумерными массивами реализован поиск большего по модулю значения в строке. После этого мы делим все значения этой строки на этот больший элемент и выводим новый массив.