|  |  |  |
| --- | --- | --- |
| Министерство образования Республики Беларусь  Учреждение образования  Белорусский Государственный Университет Информатики и Радиоэлектроники | | |
| Факультет компьютерных систем и сетей | | |
| Кафедра электронных вычислительных средств | | |
| **ОТЧЕТ**  по лабораторной работе №3  «Одномерные массивы» | | |
| Выполнил  Патюпин М. С.  Группа 250505 |  | Проверил(а)  Скиба И. Г. |
| Минск 2022 | | |

**1 ЦЕЛЬ И ЗАДАЧИ ЛАБОРАТОРНОЙ РАБОТЫ**

**1.1**  Целью ЛР является научиться разрабатывать линейные и разветвляющиеся алгоритмы и писать код на языке Си по составленному алгоритму.

**1.2** Для достижения поставленной цели необходимо решить следующие задачи:

1) Изучить лекционный материал по теме «Одномерные массивы»

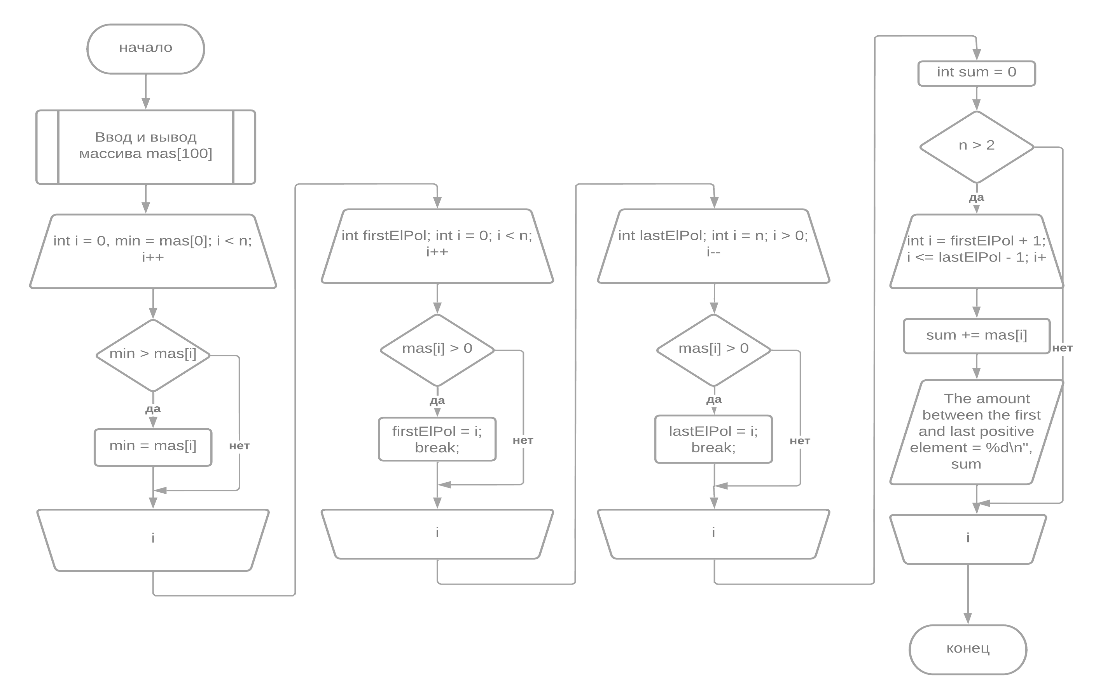
2) Выполнить следующее задание по ЛР в соответствии с вариантом №10, разработать алгоритм его реализации, запрограммировать с использованием языка «Си», отладить и представить результат работы компьютерной программы, а также блок-схему программы.

**2 РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ**

**2.1 Результат выполнения программы 1**

В одномерном массиве, состоящем из п вещественных элементов, вычислить:  
- минимальный элемент массива;  
- сумму элементов массива, расположенных между первым и последним положительными элементами.

**2.1.1** Блок-схема алгоритма смешанного представлена на рисунке 1.1.

****

**2.1.2** Листинг компьютерной программы:

#include <stdio.h>

#include "Header.h"

#include <stdlib.h>

int main() {

float mas[100];

int n;

printf("Enter the number of elements of the array");

if (scanf\_s("%d", &n) != 1 or n < 0) {

printf("Error"); return 0;

}

int fl;

printf("If you want to fill in the array manually - enter 0, if you want autofill - 1\n ");

scanf\_s("%d", &fl);

if (fl == 0) {

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

printf("%d element = ", i + 1);

scanf\_s("%f", &mas[i]);

}

}

else {

for (int i = 0, flZnak = 0; i < n; i++) {

float znah\_i = 0;

znah\_i = rand() % 200;

flZnak = rand() % 2;

if (flZnak == 0)

mas[i] = znah\_i;

else

mas[i] =0 - znah\_i;

}

}

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

printf("%d elemetn = %2.f\n", i+1, mas[i]);

}

int min = mas[0];

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

if (min > mas[i])

min = mas[i];

}

printf("\nMin element = %d\n", min);

int firstElPol;

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

if (mas[i] > 0) { firstElPol = i; break; }

}

int lastElPol;

for (int i = n; i > 0; i--) {

if (mas[i] > 0) { lastElPol = i; break; }

}

int sum = 0;

if (n > 2) {

for (int i = firstElPol + 1; i <= lastElPol - 1; i++) {

sum += mas[i];

}

printf("Sum megdy ferst and last pologitelnimi elementsmi = %d\n", sum);

}

return 0;

}

**2.1.3** Результат выполнения компьютерной программы, представлен на рисунке 1.2.

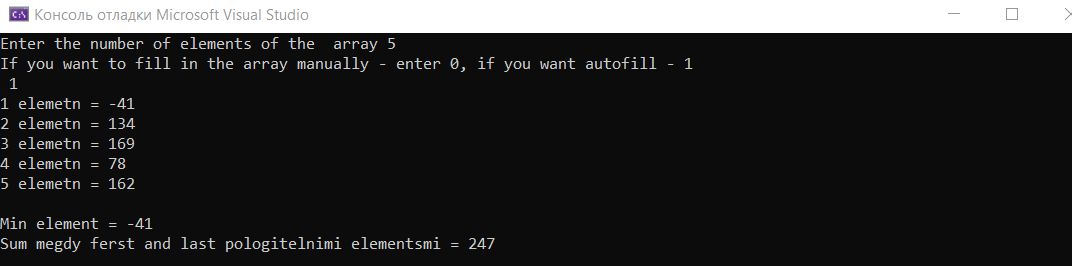
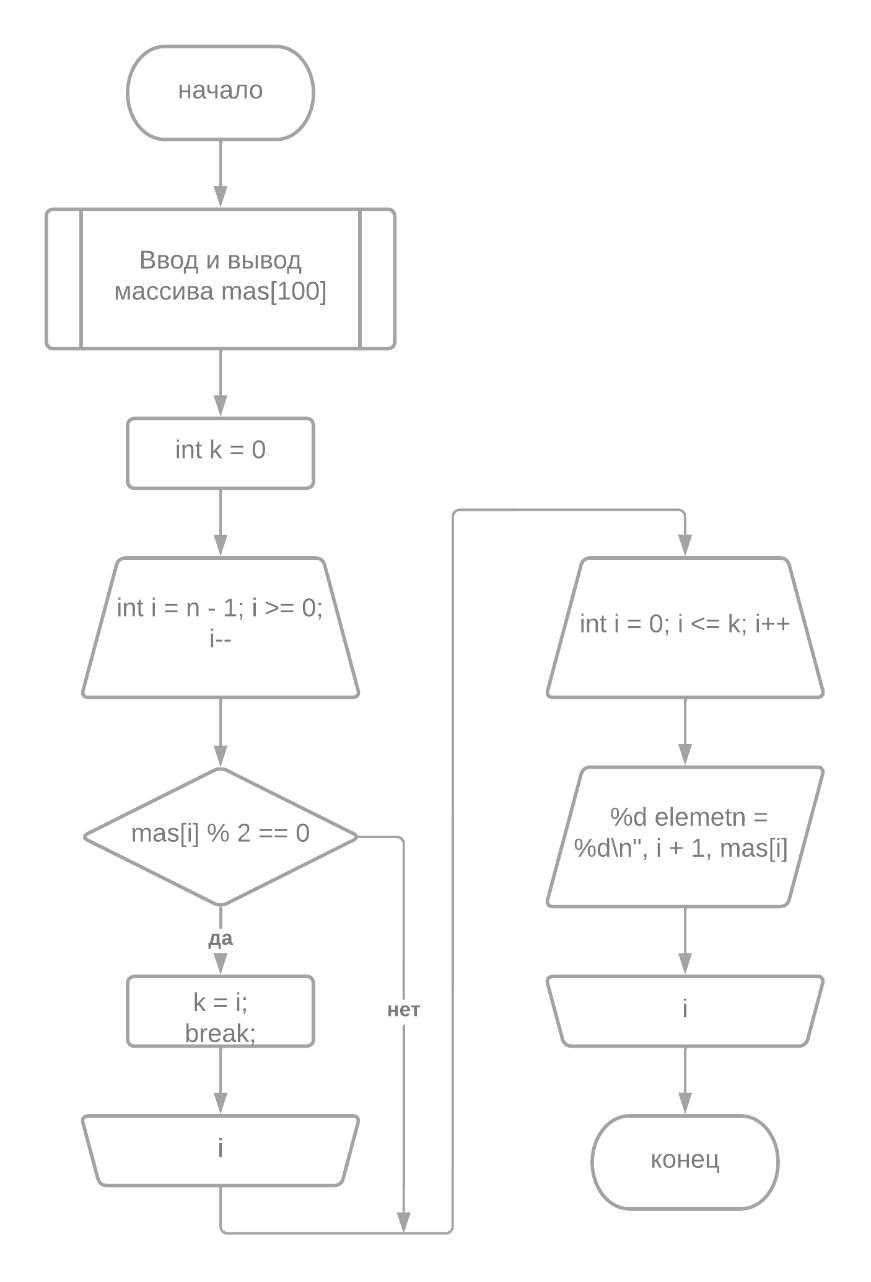


Рисунок 1.2 - Результат выполнения программы

**2.2 Результат выполнения программы 2**

В массиве из n элементов удалить все элементы расположенные после последнего четного

**2.2.1** Блок-схема алгоритма смешанного типа предоставлена на рисунке 2.



Рисунок

**2.1.2** Листинг компьютерной программы:

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int main() {

srand(time(NULL));

int mas[100] = { 0 };

int n;

printf("Enter the number of elements of the array ");

if (scanf\_s("%d", &n) != 1 or n < 0) {

printf("Error"); return 0;

}

int fl;

printf("If you want to fill in the array manually - enter 0, if you want autofill - 1\n ");

scanf\_s("%d", &fl);

if (fl == 0) {

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

printf("%d element = ", i + 1);

scanf\_s("%d", &mas[i]);

}

}

else {

for (int i = 0, flZnak = 0; i < n; i++) {

int znah\_i = 0;

znah\_i = rand() % 200;

flZnak = rand() % 2;

if (flZnak == 0)

mas[i] = znah\_i;

else

mas[i] = 0 - znah\_i;

}

}

printf("\n");

int k = 0;

for (int i = n - 1; i >= 0; i--) {

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

k = i;

break;

}

}

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

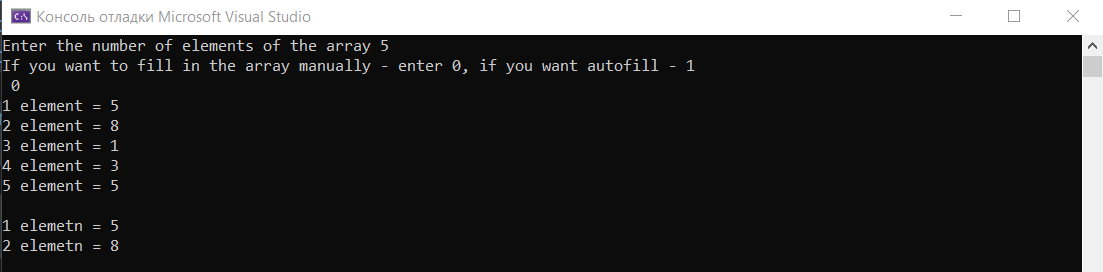
printf("%d elemetn = %d\n", i + 1, mas[i]);

}

return 0;

}

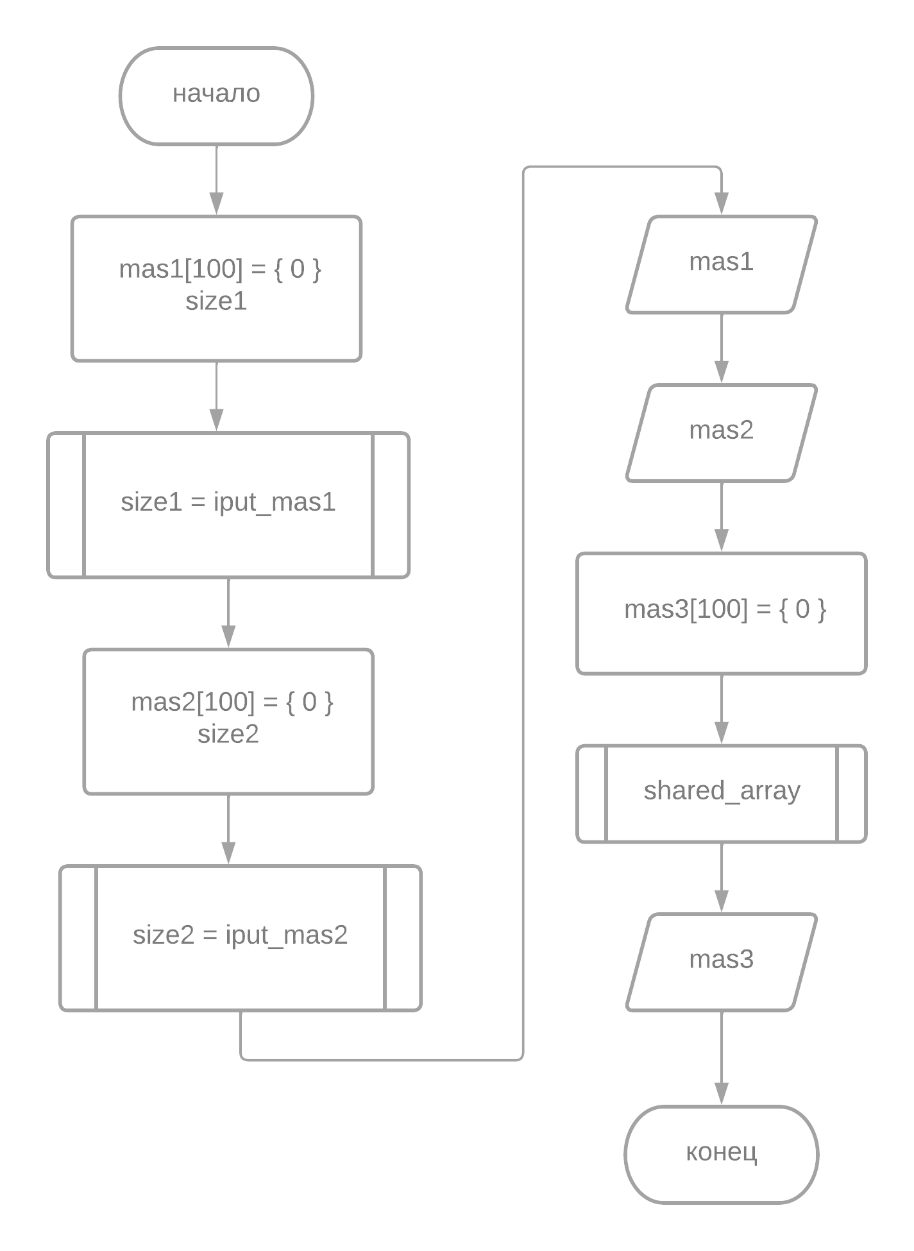
**2.2.3** Результат выполнения компьютерной программы, представлен на рисунке 3.



Рисунок

**2.3 Результат выполнения программы 3**

**2.3.1** Блок-схема алгоритма смешанного типа предоставлена на рисунке 4:

****

Рисунок

**2.3.2** Листинг компьютерной программы:

#include <time.h>

#include <stdio.h>

#include <stdlib.h>

int iput\_mas1(int\* mas1) {

int size1 = 100;

printf("Enter the number of elements of the first array ");

while (scanf\_s("%d", &size1) != 1 or 0 > size1 > 100) {

printf("Try agin\a\n");

}

int fl1;

printf("If you want to fill in the array manually - enter 0, if you want autofill - 1\n ");

scanf\_s("%d", &fl1);

if (fl1 == 0) {

int input, i = 0;

while (i < size1) {

do {

printf("%d element = ", i + 1);

scanf\_s("%d", &input);

}

while ((i > 0) && (input >= mas1[i - 1]));

mas1[i] = input;

i++;

}

}

else {

int input, i = 0;

while (i < size1) {

do {

input = rand() % 200;

} while ((i > 0) && (input >= mas1[i - 1]));

mas1[i] = input;

i++;

}

}

return size1;

}

int iput\_mas2(int\* mas2) {

int size2 = 100;

printf("Enter the number of elements of the second array ");

while (scanf\_s("%d", &size2) != 1 and 0 < size2 < 100) {

printf("Try agin\a\n");

}

int fl1;

printf("If you want to fill in the array manually - enter 0, if you want autofill - 1\n ");

scanf\_s("%d", &fl1);

if (fl1 == 0) {

int input, i = 0;

while (i < size2) {

do {

printf("%d element = ", i + 1);

scanf\_s("%d", &input);

} while ((i > 0) && (input <= mas2[i - 1]));

mas2[i] = input;

i++;

}

}

else {

int input, i = 0;

while (i < size2) {

do {

input = rand() % 200;

} while ((i > 0) && (input <= mas2[i - 1]));

mas2[i] = input;

i++;

}

}

return size2;

}

void shared\_array(int size1, int size2, int\* mas1, int\* mas2, int\* mas3) {

int size3 = size1 + size2;

int i3 = 0;

int i1 = 0, i2 = size2 - 1;

while (i1 < size1 && i2 > -1) {

if (mas1[i1] >= mas2[i2]) {

mas3[i3] = mas1[i1];

i1++;

}

else {

mas3[i3] = mas2[i2];

i2--;

}

i3++;

}

//Если один из массивов закончился, то заполняем оставшегося

while (i1 < size1 && i2 > -1) {

if (mas1[i1] >= mas2[i2]) {

mas3[i3] = mas1[i1];

i1++;

}

else {

mas3[i3] = mas2[i2];

i2--;

}

i3++;

}

if (i1 == size1 && (i2 >= 0)) {

while (i2 != 0) {

mas3[size1 + (size2 - i2) + 1] = mas2[i2];

i2--;

}

}

if (i2 < 0 && (i1 < size1)) {

while (i1 != size1) {

mas3[size2 + i1] = mas1[i1];

i1++;

}

}

}

int main() {

srand(time(NULL));

int mas1[100] = { 0 };

int size1 = iput\_mas1(mas1);

int mas2[100] = { 0 };

int size2 = iput\_mas2(mas2);

printf("\n");

for (int i = 0; i < size1; i++) {

printf("mas1[%d] = %d\n", i +1, mas1[i]);

}

printf("\n");

for (int i = 0; i < size2; i++) {

printf("mas2[%d] = %d\n", i +1, mas2[i]);

}

int mas3[200] = {0};

shared\_array(size1, size2, mas1, mas2, mas3);

printf("\n");

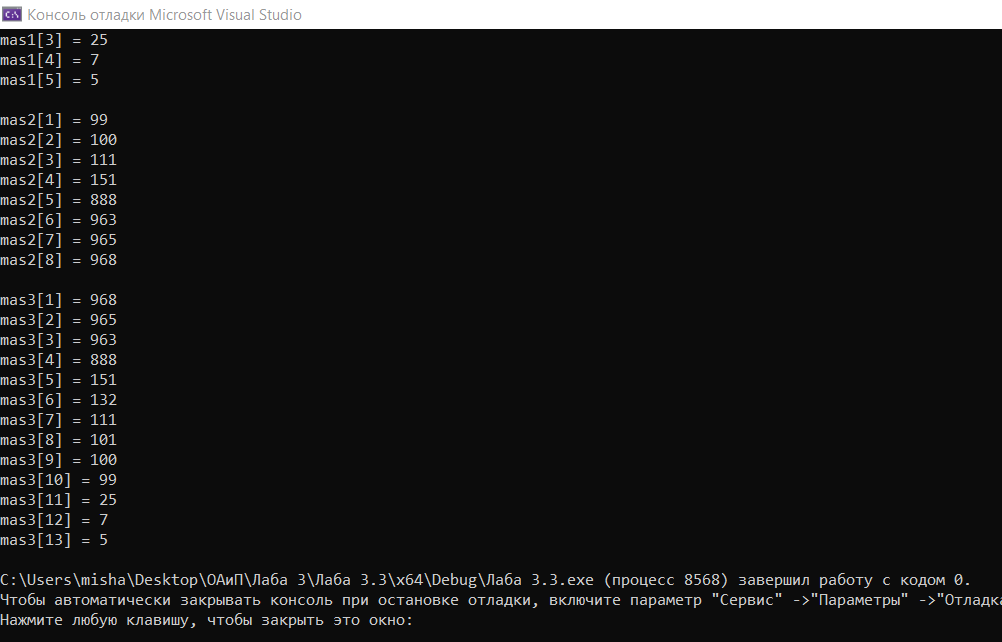
for (int i = 0; i < size1 + size2; i++) {

printf("mas3[%d] = %d\n", i + 1, mas3[i]);

}

return 0;

}

**2.3.3** Результат выполнения компьютерной программы, представлен на рисунке 5.

Рисунок

**2.4 Выводы по результатам выполнения ЛР**

В результате выполнения ЛР я научился составлять линейные и разветвляющиеся алгоритмы, а так же писать код на языке Си по составленному алгоритму

**3 СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

[1] Лекционный материал по теме «Введение в Си» .

[2] «Как программировать на С» Х.М. Дейтел, П. Дж. Дейтел