Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное учреждение высшего образования

«Пермский национальный исследовательский политехнический университет»

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

О Т Ч Ё Т

Объектно-ориентированное программирование

Семестр 3

Лабораторная работа 5

Выполнил

студент группы РИС-22-2б

Баяндин К.С.

Проверила

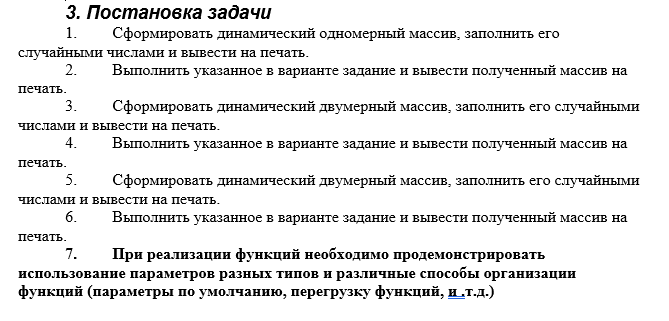
доцент кафедры ИТАС

Викентьева О.Л.

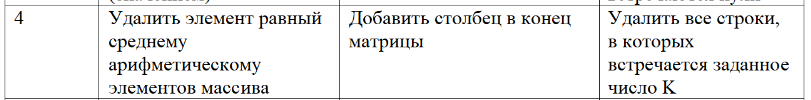
Пермь, 2023

**1. Постановка задачи**

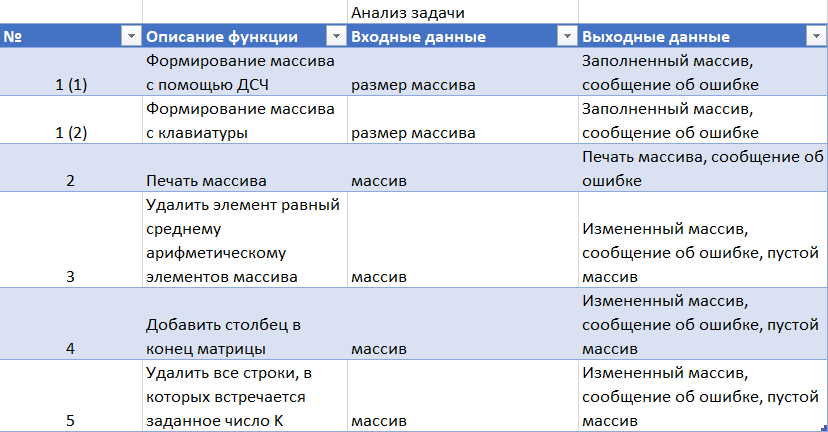
**1.1. Общая постановка задачи**

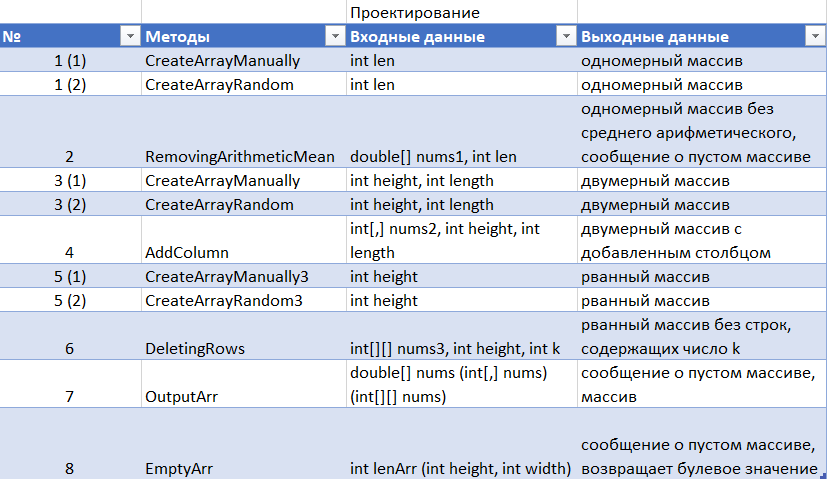


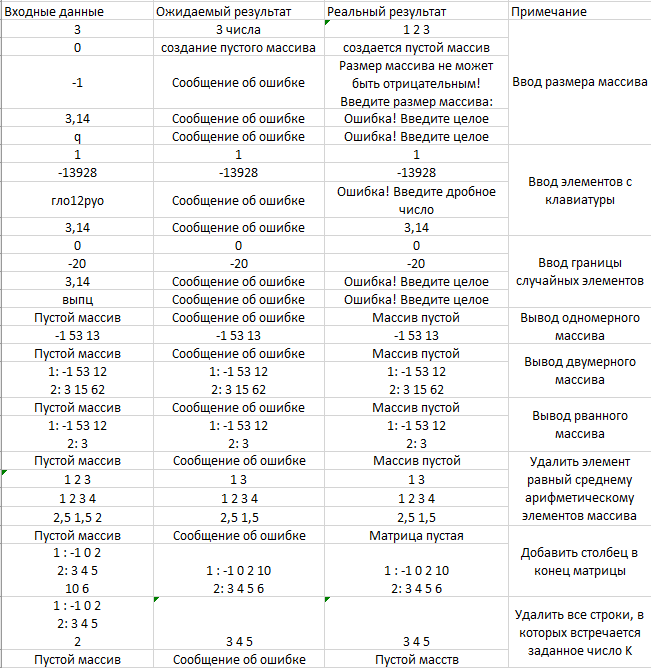
**1.2. Постановка задачи для варианта 4**

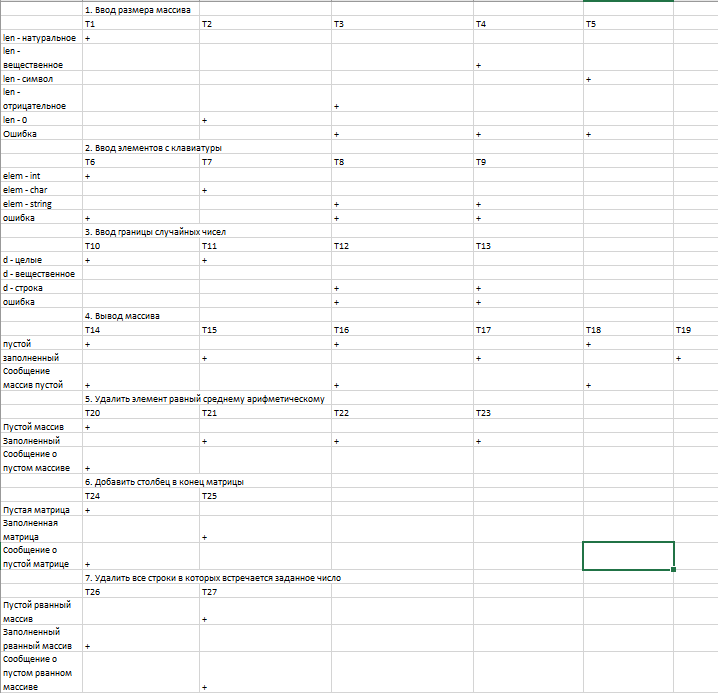


**Анализ задачи**









**Алгоритм работы программы**

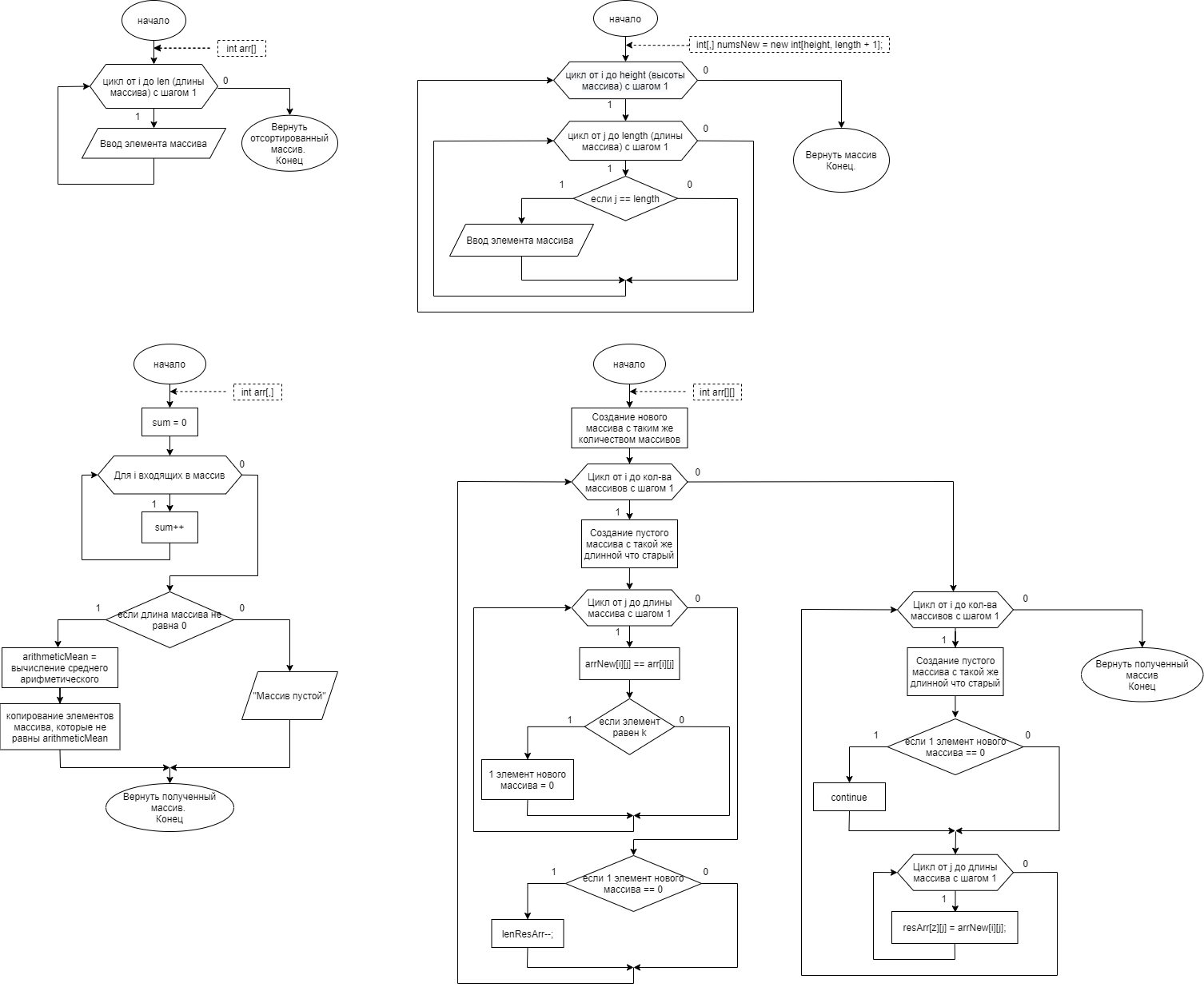


Рисунок 1 – функция создания массива

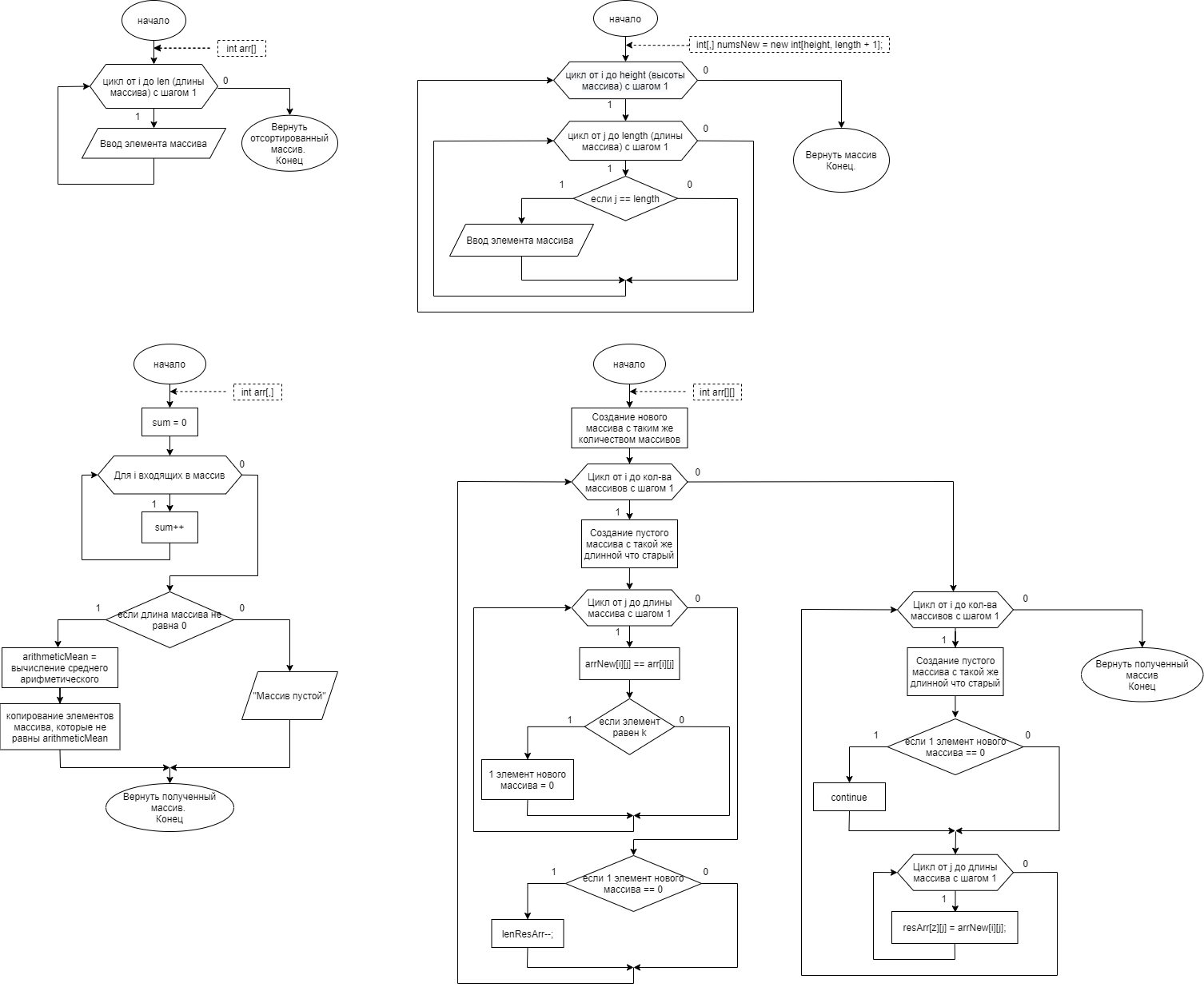


Рисунок 2 – функция удаления среднего арифметического

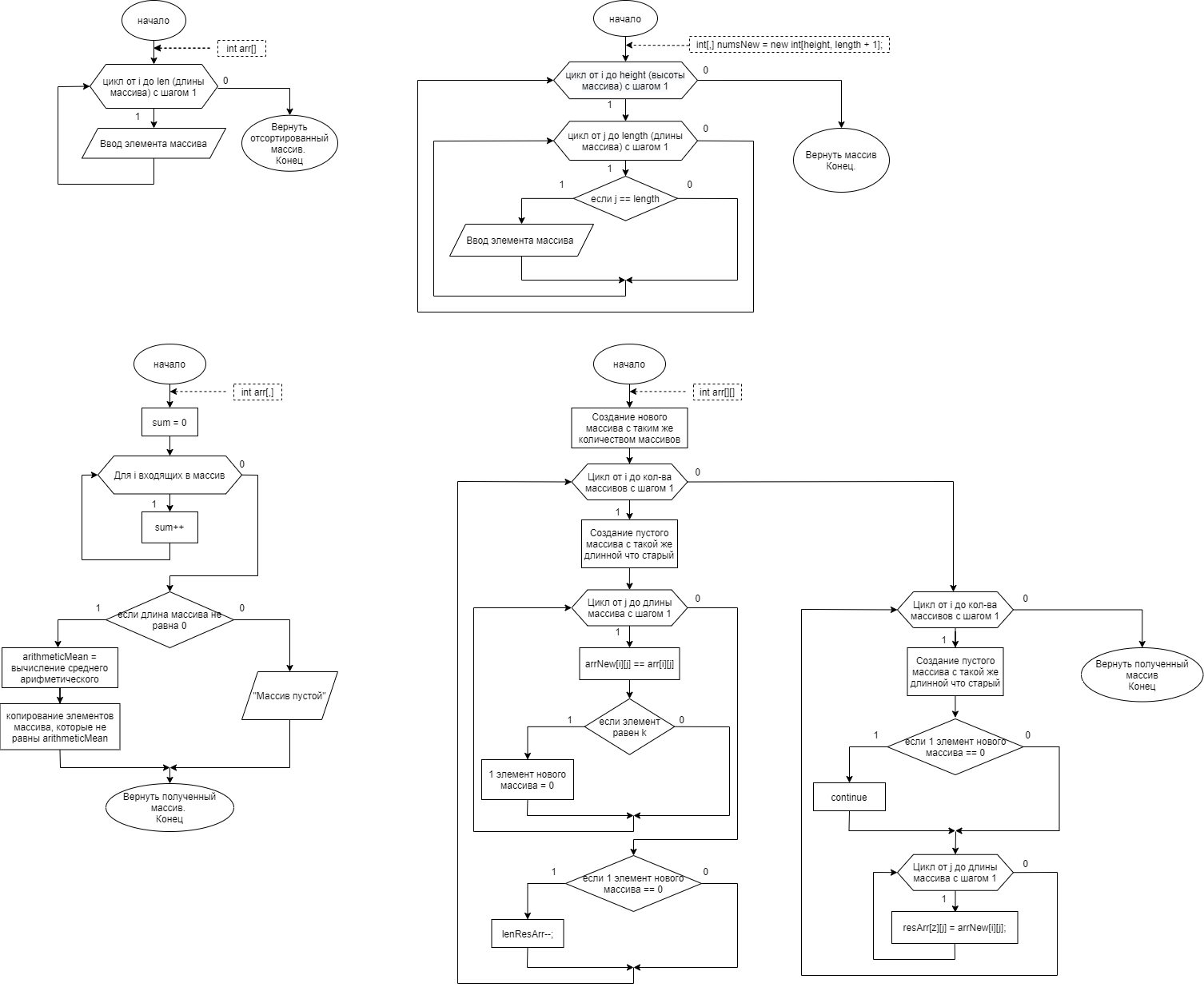


Рисунок 3 – функция вставки столбца в матрицу

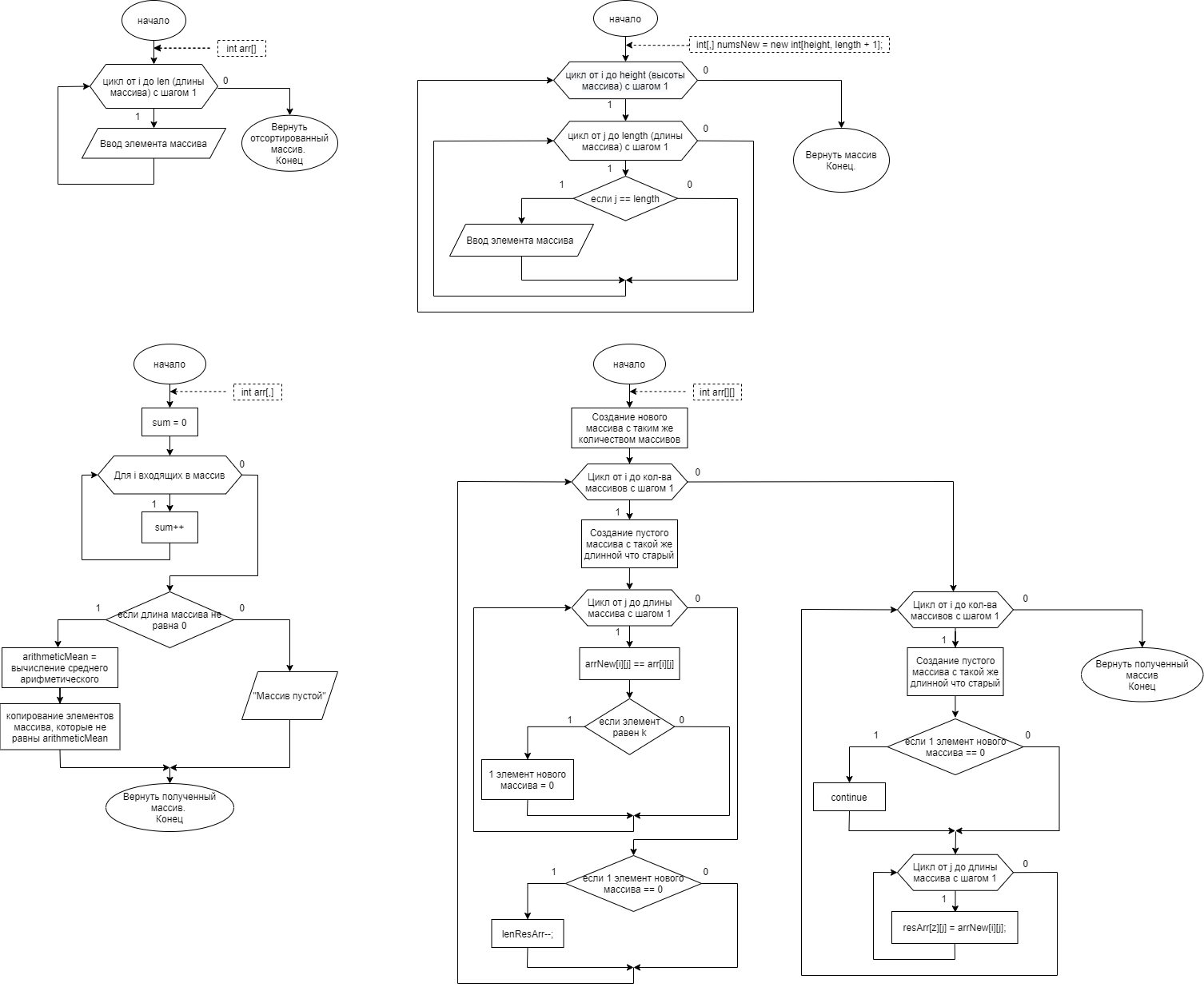


Рисунок 4 – функция удаления строк, содержащих K элемент

**Листинг программы**

using System;

using System.Linq;

namespace Lab\_OOP\_5

{

class Program

{

//заполнение одномерного массива вручную

static double[] CreateArrayManually(int len)

{

double[] nums1 = new double[len];

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

nums1[i] = InputDouble("Введите " + (i + 1) + " элемент массива: ");

Array.Sort(nums1);

return nums1;

}

//заполнение одномерного массива случайными элементами

static double[] CreateArrayRandom(int len)

{

double[] nums1 = new double[len];

Console.WriteLine("Введите диапазон задания случайных чисел: ");

int left = InputInt("Введите левую границу: ");

int right = InputInt("Введите правую границу: ");

Random rnd = new Random();

byte[] bytes = new byte[100];

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

{

rnd.NextBytes(bytes);

nums1[i] = rnd.Next(left, right);

}

Array.Sort(nums1);

return nums1;

}

//удаление среднего арифметического из массива

static double[] RemovingArithmeticMean(double[] nums1, int len)

{

double sum = 0;

foreach (double i in nums1)

sum += i;

if (len != 0)

{

double arithmeticMean = sum / len;

nums1 = nums1.Where(s => s != arithmeticMean).ToArray();

}

else

Console.WriteLine("Массив пустой!");

return nums1;

}

//заполнение двумерного массива вручную

static int[,] CreateArrayManually(int height, int length)

{

int[,] nums2 = new int[height, length];

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

{

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

nums2[i, j] = InputInt("Введите " + (j + 1) + " элемент массива: ");

}

return nums2;

}

//заполнение двумерного массива случайными элементами

static int[,] CreateArrayRandom(int height, int length)

{

int[,] nums2 = new int[height, length];

Console.WriteLine("Введите диапазон задания случайных чисел: ");

int left = InputInt("Введите левую границу: ");

int right = InputInt("Введите правую границу: ");

Random rnd = new Random();

byte[] bytes = new byte[100];

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

{

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

{

rnd.NextBytes(bytes);

nums2[i, j] = rnd.Next(left, right);

}

}

return nums2;

}

//добавление столбца в конец матрицы (вручную)

static int[,] AddColumnManually(int[,] nums2, int height, int length)

{

int[,] numsNew = new int[height, length + 1];

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

{

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

numsNew[i, j] = (j == length) ? (InputInt("Введите " + (i + 1) + " элемент столбца: ")) : (nums2[i, j]);

}

return numsNew;

}

//добавление столбца в конец матрицы (рандомно)

static int[,] AddColumnRandom(int[,] nums2, int height, int length)

{

int[,] numsNew = new int[height, length + 1];

int left = InputInt("Введите левую границу: ");

int right = InputInt("Введите правую границу: ");

Random rnd = new Random();

byte[] bytes = new byte[100];

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

{

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

{

rnd.NextBytes(bytes);

numsNew[i, j] = (j == length) ? (rnd.Next(left, right)) : (nums2[i, j]);

}

}

return numsNew;

}

//заполнение рванного массива вручную

static int[][] CreateArrayManually3(int height)

{

int[][] nums3 = new int[height][];

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

{

int length = InputInt("Введите размер " + (i + 1) + " массива: ");

nums3[i] = new int[length];

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

nums3[i][j] = InputInt("Введите " + (j + 1) + " элемент массива: ");

}

return nums3;

}

//заполнение рванного массива рандомно

static int[][] CreateArrayRandom3(int height)

{

int[][] nums3 = new int[height][];

Console.WriteLine("Введите диапазон задания случайных чисел: ");

int left = InputInt("Введите левую границу: ");

int right = InputInt("Введите правую границу: ");

Random rnd = new Random();

byte[] bytes = new byte[100];

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

{

int length = InputInt("Введите размер " + (i + 1) + " массива: ");

nums3[i] = new int[length];

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

{

rnd.NextBytes(bytes);

nums3[i][j] = rnd.Next(left, right);

}

}

return nums3;

}

//удаление строк, в которых встречается число K

static int[][] DeletingRows(int[][] nums3, int height, int k)

{

int[][] numsNew = new int[height][];

int lenResArr = height;

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

{

int length = nums3[i].Length;

numsNew[i] = new int[length];

if (length == 0)

break;

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

{

numsNew[i][j] = nums3[i][j];

if (nums3[i][j] == k)

numsNew[i][0] = 0;

}

if (numsNew[i][0] == 0)

lenResArr--;

}

int[][] resArr = new int[lenResArr][];

int z = 0;

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

{

int length = numsNew[i].Length;

resArr[z] = new int[length];

if (length == 0)

break;

if (numsNew[i][0] == 0)

continue;

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

resArr[z][j] = numsNew[i][j];

z++;

}

return resArr;

}

//проверка на ввод целого числа

static int InputInt(string message)

{

int number;

Console.Write(message);

if (!int.TryParse(Console.ReadLine(), out number))

{

do

{

Console.WriteLine("Ошибка!");

Console.WriteLine("Введите число типа int");

Console.Write(message);

} while (!int.TryParse(Console.ReadLine(), out number));

}

return number;

}

//проверка на ввод дробного числа

static double InputDouble(string message)

{

double number;

Console.Write(message);

if (!double.TryParse(Console.ReadLine(), out number))

{

do

{

Console.WriteLine("Ошибка!");

Console.WriteLine("Введите число типа double");

Console.Write(message);

} while (!double.TryParse(Console.ReadLine(), out number));

}

return number;

}

//проверка на пустой одномерный массив

static bool EmptyArr(int lenArr)

{

if (lenArr == 0)

Console.WriteLine("Массив пустой!");

return (lenArr == 0);

}

//проверка на пустую матрицу

static bool EmptyArr(int height, int width)

{

if (height == 0 || width == 0)

Console.WriteLine("Матрица пустая!");

return (height == 0 || width == 0);

}

//вывод на экран одномерного массива

static void OutputArr(double[] nums)

{

for (int i = 0; i < nums.Length; i++)

Console.Write(nums[i] + " ");

Console.WriteLine();

}

//вывод на экран матрицы

static void OutputArr(int[,] nums)

{

int rows = nums.GetUpperBound(0) + 1; // количество строк

int columns = nums.Length / rows; // количество столбцов

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

{

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

Console.Write(nums[i, j] + " ");

Console.WriteLine();

}

Console.WriteLine();

}

//вывод на экран рванного массива

static void OutputArr(int[][] nums)

{

if (nums == null)

{

Console.WriteLine("Пусто!");

return;

}

foreach (int[] row in nums)

{

if (row == null)

{

Console.WriteLine("Пусто!");

return;

}

foreach (int number in row)

Console.Write(number + " ");

Console.WriteLine();

}

}

static void Main()

{

string[] menu =

{

"1. Создание одномерного массива (заполнение вручную)\n",

"2. Создание одномерного массива (заполнение рандомно)\n",

"3. Удаление среднего арифметического из одномерного массива\n\n",

"4. Создание двумерного массива (заполнение вручную)\n",

"5. Создание двумерного массива (заполнение рандомно)\n",

"6. Добавление столбца в конец матрицы (заполнение вручную)\n",

"7. Добавление столбца в конец матрицы (заполнение рандомно)\n\n",

"8. Создание рванного массива (заполнение вручную)\n",

"9. Создание рванного массива (заполнение рандомно)\n",

"10. Удаление из рванного массива строк, содержащие элемент K\n",

"0. Выход из программы\n"

};

int lenArr = 0;

int height = 0;

int width = 0;

int numArr = 0;

double[] nums1 = new double[lenArr];

int[,] nums2 = new int[height, lenArr];

int[][] nums3 = new int[numArr][];

bool isWork = true;

while (isWork)

{

//вывод меню

for (int i = 0; i < menu.Length; i++)

{

Console.WriteLine(menu[i]);

}

int command = InputInt("\nВведите команду: ");

switch (command)

{

case 1: //Создание одномерного массива(заполнение вручную)

lenArr = InputInt("\nВведите размер массива: ");

while (lenArr < 0)

lenArr = InputInt("\nРазмер массива не может быть отрицательным! Введите размер массива: ");

nums1 = new double[lenArr];

nums1 = CreateArrayManually(lenArr);

OutputArr(nums1);

break;

case 2: //Создание одномерного массива (заполнение рандомно)

lenArr = InputInt("\nВведите размер массива: ");

nums1 = new double[lenArr];

nums1 = CreateArrayRandom(lenArr);

OutputArr(nums1);

break;

case 3: //Удаление среднего арифметического из одномерного массива

nums1 = RemovingArithmeticMean(nums1, lenArr);

OutputArr(nums1);

break;

case 4: //Создание двумерного массива (заполнение вручную)

height = InputInt("\nВведите количество строк в матрице: ");

width = InputInt("\nВведите количество столбцов в матрице: ");

if (EmptyArr(height, width))

break;

nums2 = new int[height, width];

nums2 = CreateArrayManually(height, width);

OutputArr(nums2);

break;

case 5: //Создание двумерного массива (заполнение рандомно)

height = InputInt("\nВведите количество строк в матрице: ");

width = InputInt("\nВведите количество столбцов в матрице: ");

if (EmptyArr(height, width))

break;

nums2 = new int[height, width];

nums2 = CreateArrayRandom(height, width);

OutputArr(nums2);

break;

case 6: //Добавление столбца в конец матрицы (заполнение вручную)

if (EmptyArr(height, width))

break;

nums2 = AddColumnManually(nums2, height, width);

OutputArr(nums2);

break;

case 7: //Добавление столбца в конец матрицы (заполнение рандомно)

if (EmptyArr(height, width))

break;

nums2 = AddColumnRandom(nums2, height, width);

OutputArr(nums2);

break;

case 8: //Создание рванного массива (заполнение вручную)

numArr = InputInt("\nВведите количество массивов: ");

nums3 = new int[numArr][];

nums3 = CreateArrayManually3(numArr);

OutputArr(nums3);

break;

case 9: //Создание рванного массива (заполнение рандомно)

numArr = InputInt("\nВведите количество массивов: ");

nums3 = new int[numArr][];

nums3 = CreateArrayRandom3(numArr);

OutputArr(nums3);

break;

case 10: //Удаление из рванного массива строк, содержащие элемент K

if (EmptyArr(numArr))

break;

int k = InputInt("Введите k (удаление всех строк, содержащих k): ");

nums3 = DeletingRows(nums3, numArr, k);

OutputArr(nums3);

break;

case 0: //Выход из программы

isWork = false;

break;

default:

Console.WriteLine("Неправильно введена команда!");

break;

}

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.ReadKey();

}

}

}

}