МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Костромской государственный университет»

(КГУ)

Институт физико-математических и естественных наук

Кафедра защиты информации

Направление подготовки: 10.03.01

Информационная безопасность

Дисциплина: Языки и методы программирования

(наименование дисциплины)

**ЛАБОРАТОРНАЯ РАБОТА №7**

«Сравнение алгоритмов сортировки»

Выполнил студент: Сотов Сергей Сергеевич

Группа 20-ИБбо-6

Проверил: доцент кафедры "Защиты информации", кандидат технических наук

Мозохин Александр Евгеньевич

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_

Кострома

2020

Содержание

[Цели и задачи 1](#_Toc54519940)

[Первая часть задачи 2](#_Toc54519941)

[Код 2](#_Toc54519942)

[Блок-схема 3](#_Toc54519943)

[Вторая часть задачи 5](#_Toc54519944)

[Код 5](#_Toc54519945)

[Блок-схема 6](#_Toc54519946)

[Вывод 6](#_Toc54519947)

# Цели и задачи

Цель: Применение простых алгоритмов сортировки при работе с массивами данных.

Задача: Разработать программу сортировки одномерных массивов:

1. Дан массив из 20 элементов. Заполнить массив случайными числами и реализовать сортировку массива тремя различными методами с подсчетом кол-ва итераций основного цикла;
2. Дан массив русских имен из 10 элементов. Отсортировать этот массив любым методом сортировки.

# Первая часть задачи

## Код

Пузырьковый метод: благодаря следующему коду сравниваются два числа, стоящие рядом, и если первое больше второго, то они меняются местами:

while (flag != true)

{

flag = true;

for (int i = 0; i < massiv.Length - 1; i++)

{

if (massiv[i] > massiv[i + 1])

{

dop = massiv[i];

massiv[i] = massiv[i + 1];

massiv[i + 1] = dop;

flag = false;

}

}

}

Метод вставки: алгоритм берет число, находит ему подходящее место в массиве, сравнивая его со всеми числами:

for (int i = 1; i < massiv2.Length; i++)

{

g = i;

for (int j = i - 1; j >= 0; j--)

{

if (massiv2[j] > massiv2[g])

{

dop = massiv2[g];

massiv2[g] = massiv2[j];

massiv2[j] = dop;

g--;

}

}

}

Метод выбора: алгоритм находит наибольшее число и ставит его в конец массива:

while (k > 0)

{

level = 0;

for (int i = 1; i < k+1; i++)

{

if (massiv3[i] >= massiv3[level])

{

level = i;

}

}

need = massiv3[k];

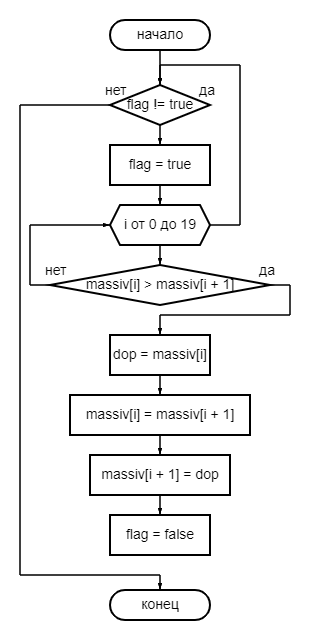
massiv3[k] = massiv3[level];

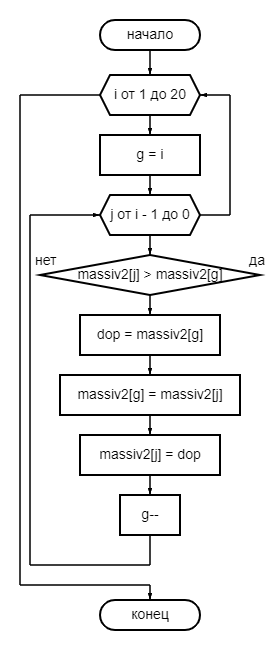
massiv3[level] = need;

k--;

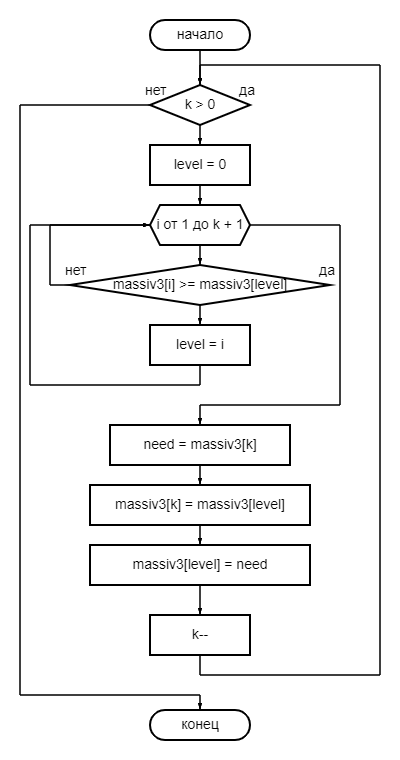
}

## Блок-схема

Пузырьковый метод:

Метод вставки:

Метод выбора:



# Вторая часть задачи

## Код

Чтобы отсортировать массив имен я взял пузырьковый метод. Сортировка происходит по первой букве имени, а точнее по ее коду в таблице ASKII.

while (flag != true)

{

flag = true;

for (int i = 0; i < names.Length - 1; i++)

{

if ((int)names[i][0] > (int)names[i + 1][0])

{

dop2 = names[i];

names[i] = names[i + 1];

names[i + 1] = dop2;

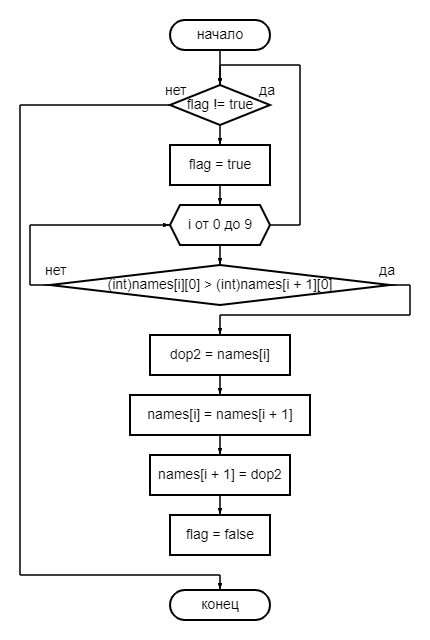
flag = false;

}

}

}

## Блок-схема



# Вывод

Разработал программу сортировки одномерных массивов, применяя простые алгоритмы сортировки.