Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №1**

**«ИЗУЧЕНИЕ ОДНОМЕРНЫХ МАССИВОВ И СТРОК»**

**ПО ДИСЦИПЛИНЕ «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»**

Выполнил: студент учебной группы

ИСПк-204-52-00

Вейс Роман Михайлович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

В отчете должны отображаться:

1. Цель работы
2. Формулировка задания (с вариантом)
3. Описание алгоритма
4. Схема алгоритма с комментариями
5. Код программы
6. Результат выполнения программы
7. Вывод

Цель работы: получить базовые навыки работы с одномерными массивами, освоить принципы работы со строками как с частными случаями одномерных массивов

Формулировка задания: 1. Написать программу, выполняющую поиск среднеарифметического в заданном массиве.

2. Написать программу, определяющую количество уникальных подстрок из 3 символов, входящих в эту строку.

3. Все данные вводятся с клавиатуры. При выполнении второго пункта запрещается использовать стандартные функции для работы со строками (за исключением функции определения длины строки).

Вариант 5.

Описание алгоритма 1 задания: Пользователь вводит кол-во элементов в массиве, присваивая длину массива, дальше вводит элементы массива, программа сразу их суммирует. Выводится массив. Вычисляется и выводится среднеарифметическое

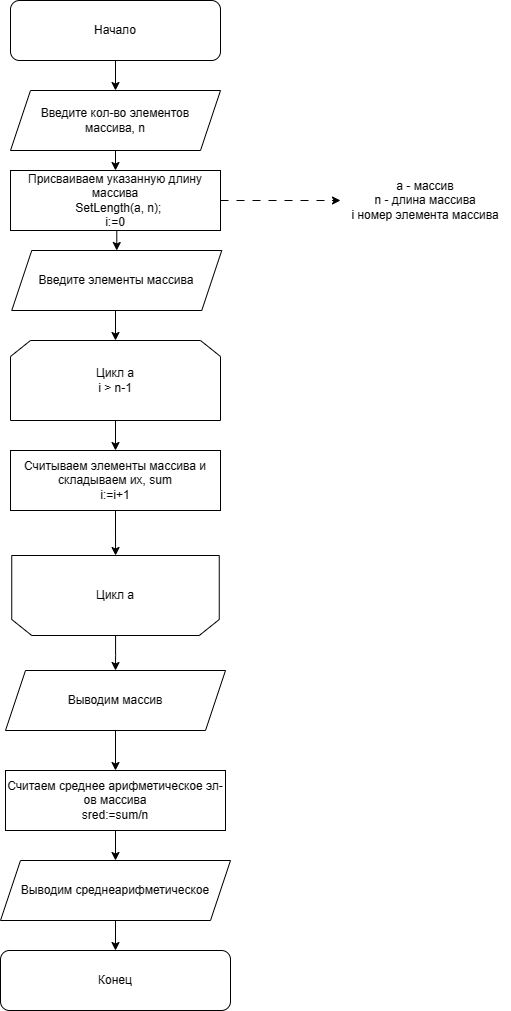


рис.1 схема-алгоритм

Код программы:

**var**

i, n, sum: int64;

sred: real;

a: **array of** int64;

**begin**

**begin**

writeln('Введите кол-во массива');

readln(n);

SetLength(a, n);

**end**;

**begin**

writeln('Введите эл-ты массива: ');

**for** i := 0 **to** n - 1 **do**

**begin**

readln(a[i]);

sum := sum + a[i];

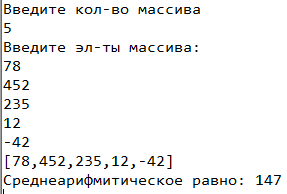
**end**;

writeln(a);

sred := sum / n;

writeln(sred);

**end**;

 рис.2 Выходные данные

Описание алгоритма 2-3 задания: Пользователь вводит задаваемую строку, из которой вычисляется количество уникальных подстрок. Программа сравнивает подстроки по 3 элемента и выводит их количество

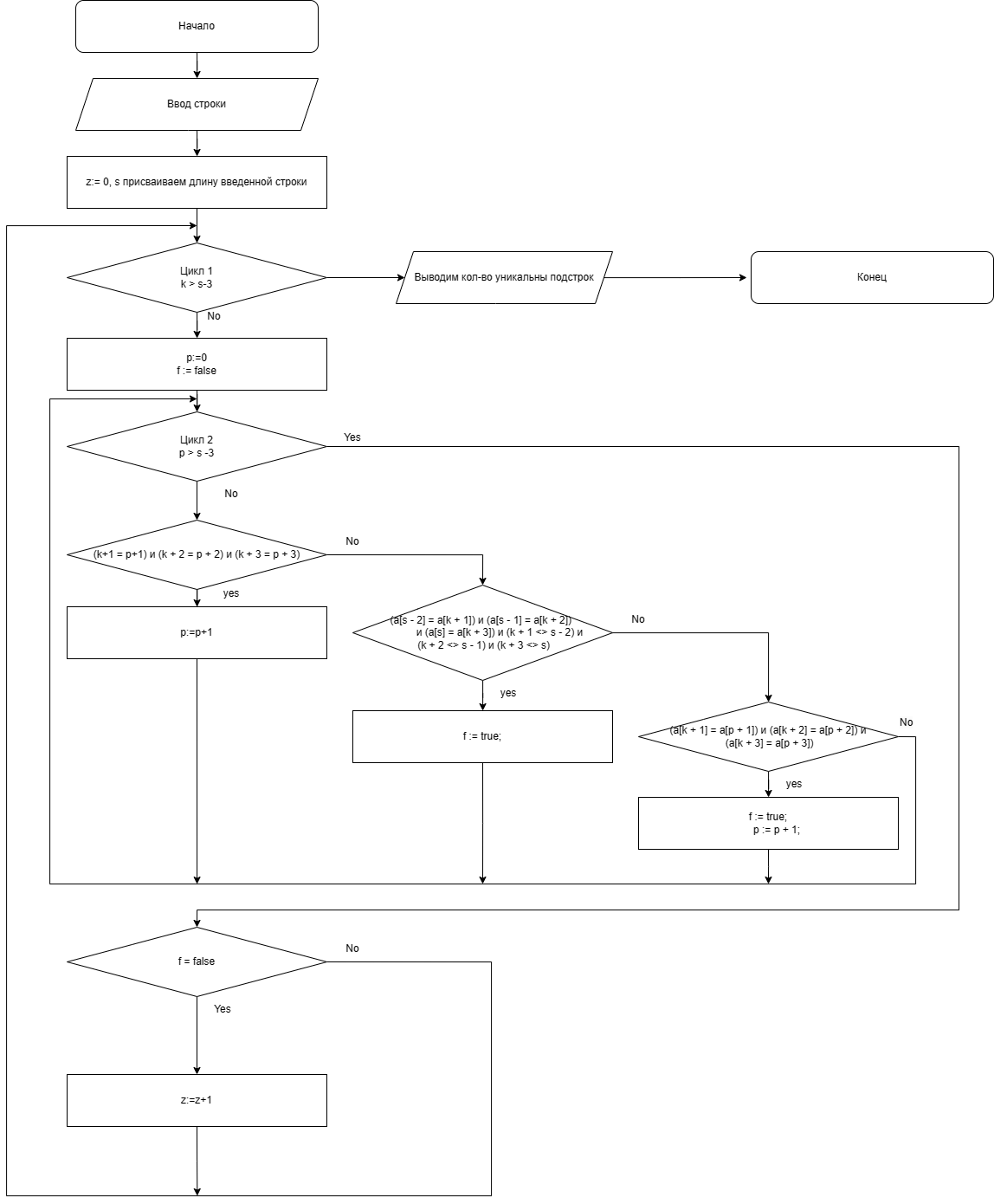


рис.3 схема-алгоритм

Код программы:

**var**

a: string;

k, s, p, z: int64;

f: Boolean;

**begin**

writeln('Введите строку');

readln(a);

z := 0; s := length(a);

**for** k := 0 **to** s - 3 **do**

**begin**

p := 0;

f := false;

**while** p < s - 3 **do**

**begin**

**if** (k + 1 = p + 1) **and** (k + 2 = p + 2) **and** (k + 3 = p + 3) **then**

p := p + 1;

**if** (a[s - 2] = a[k + 1]) **and** (a[s - 1] = a[k + 2])

**and** (a[s] = a[k + 3]) **and** (k + 1 <> s - 2) **and** (k + 2 <> s - 1) **and** (k + 3 <> s) **then**

f := true;

**if** (a[k + 1] = a[p + 1]) **and** (a[k + 2] = a[p + 2]) **and** (a[k + 3] = a[p + 3]) **then**

f := true;

p := p + 1;

**end**;

**if** f = false **then**

z := z + 1;

**end**;

write('Количество уникальных подстрок из трёх символов: ', z)

**end**.

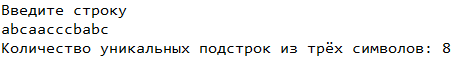


рис.4 выходные данные

Вывод:

При выполнении домашней контрольной работы мы изучали базовую структуры организации программы, основные конструкции, повторили и закрепили знания работы с языка программирования Pascal, вспомнили и закрепили знания как работает с одномерными массивами и строками. Закрепили знания как создавать схемы-алгоритмы, научились создавать к ним комментарии, научились писать отчеты.

Схема алгоритм была создана при помощи сайта и небольшой помощи преподавателя.