Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Вятский государственный университет»

Колледж ВятГУ

**ОТЧЕТ**

**ПО ДОМАШНЕЙ КОНТРОЛЬНОЙ РАБОТЕ №2**

**«ИЗУЧЕНИЕ ОДНОМЕРНЫХ МАССИВОВ И СТРОК»**

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

Выполнил: студент учебной группы

ИСПк-203-52-00

Гущин Тимофей Михайлович

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

1. **Цель работы:**

Получить базовые навыки работы с одномерными массивами, освоить принципы работы со строками как с частным случаем одномерных массивов.

1. **Формулировка задания:**

Вариант: 8

1. Написать программу, определяющую превосходит ли сумма элементов массива число k.
2. Написать программу, удаляющую из строки каждое второе вхождение заданной подстроки.
3. **Описание алгоритма:**

Задание 1.

Для решения первой задачи необходимо использовать цикл *for* для заполнения массива, а также для вычисления суммы элементов массива, конструкция *readln()* для заполнения массива и введения числа *k* с клавиатуры. Для проверки превосходства суммы элементов над числом *k* используется условный оператор *if...then…else*. В результате выполнения алгоритма будет проверяться больше, меньше или равна ли сумма элементов массив, чем число *k*. Если сумма больше числа, то на экран выводится сообщение о том, сумма элементов массива превосходит число *k*, с помощью конструкции *writeln()*, если сумма меньше, выводится сообщение “сумма элементов массива не превосходит число *k*”, а в противном случае выводится сообщение о том, что сумма элементов равна числу *k*.

Задание 2.

Для решения второй задачи используются циклы *while* для подсчета вхождений заданной подстроки в данной строке, и условные операторы *if…then…else* для нахождения и пропуска подстроки, также для увеличения переменных используется процедура *Inc()*. Результатом выполнения программы является строка без каждого второго вхождения подстроки.

1. **Схемы алгоритмов:**

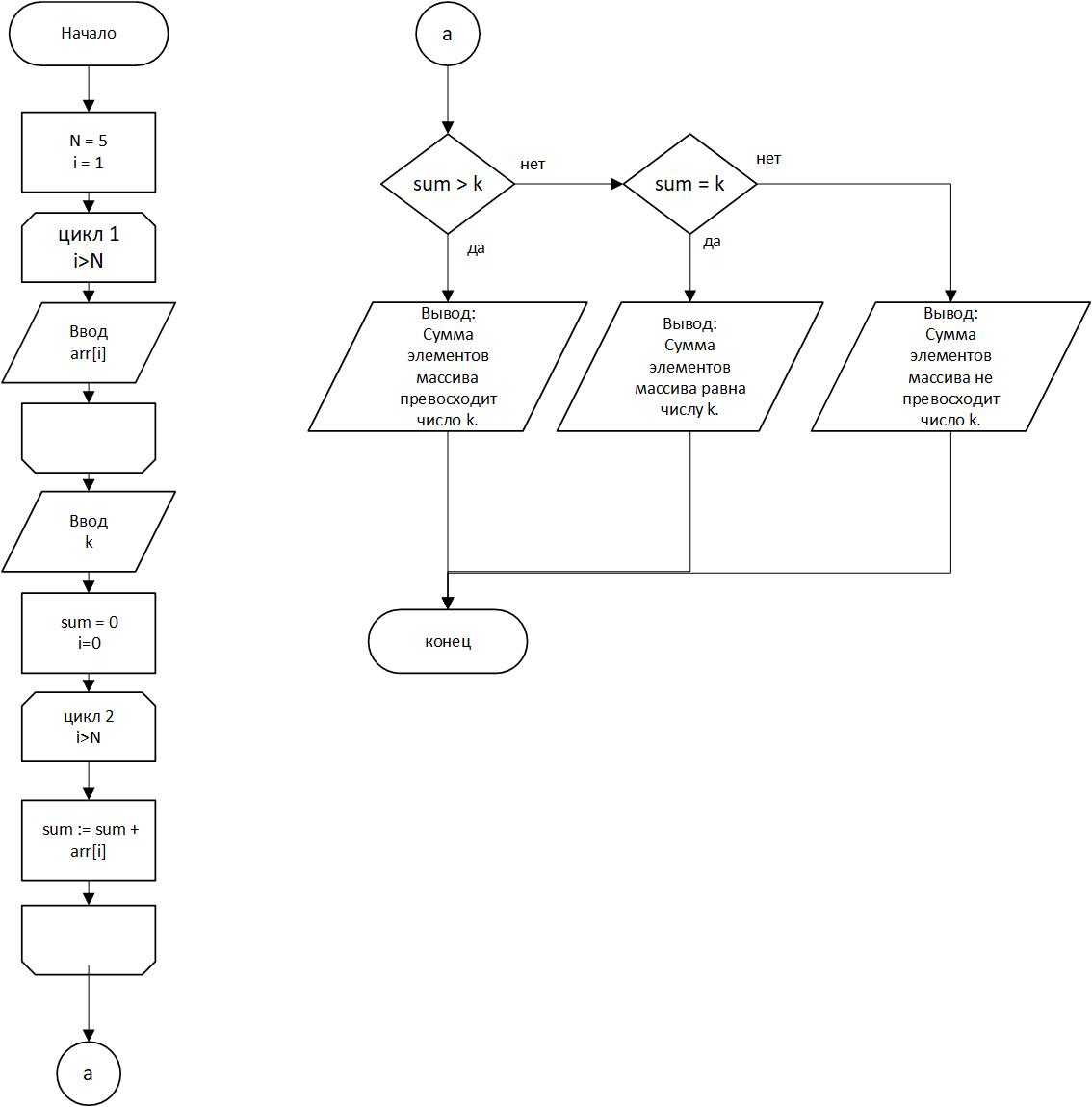
Задание 1. ****

Рис 1. – Схема алгоритма.

Задание 2.

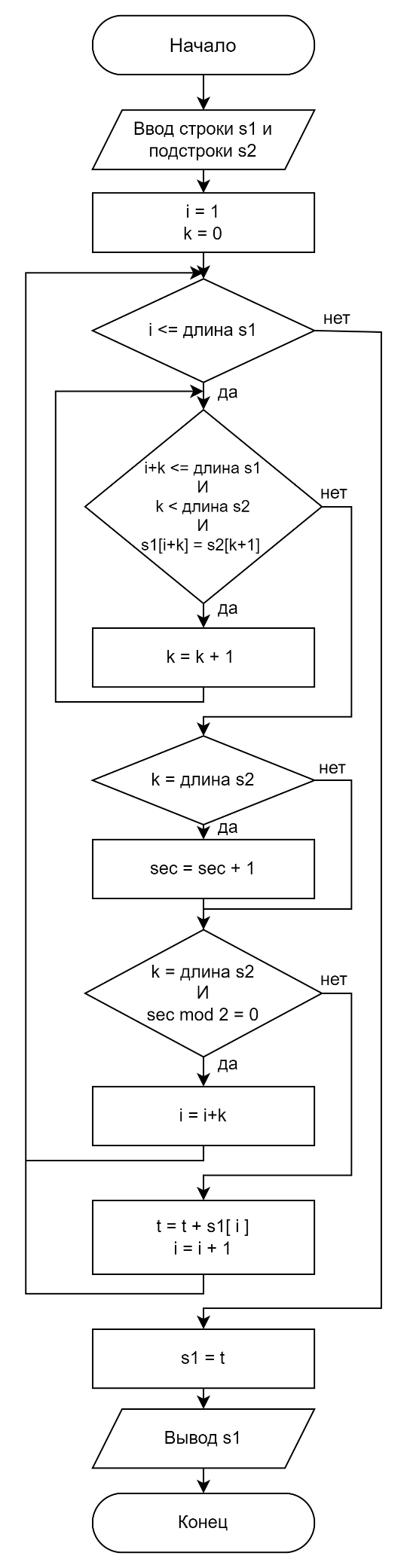


Рис 2. – Схема алгоритма.

1. **Код программы:**

Задание 1.

**program** z\_1;

**const**

N = 5;

**var**

arr: **array**[1..N] **of** Integer;

k, sum: Integer;

i: Integer;

**begin**

writeln('Введите элементы массива:');

**for** i := 1 **to** N **do**

**begin**

write('Элемент ', i, ': ');

readln(arr[i]);

**end**;

write('Введите число k: ');

readln(k);

sum := 0;

**for** i := 1 **to** N **do**

sum := sum + arr[i];

**if** sum > k **then**

writeln('Сумма элементов массива превосходит число k.')

**else if** sum = k **then**

writeln('Сумма элементов массива равна числу k.')

**else**

writeln('Сумма элементов массива не превосходит число k.');

**end**.

Задание 2.

**program** z\_2;

**var**

s1, s2, t: string;

i, j, k, sec: integer;

**begin**

writeln('Введите строку: ');

readln(s1);

writeln('Введите подстроку: ');

readln(s2);

i:=1;

**while** i <= length(s1) **do**

**begin**

k:=0;

**while** ((i+k) <= length(s1)) **and** (k < length(s2)) **and** (s1[i+k] = s2[k+1]) **do** inc(k);

**if** k = length(s2) **then** inc(sec);

**if** (k = length(s2)) **and** (sec **mod** 2 = 0) **then** inc(i, k)

**else begin**

t+=s1[i];

inc(i);

**end**;

**end**;

s1:=t;

writeln(s1);

**end**.

1. **Результат выполнения программы:**

Задание 1.

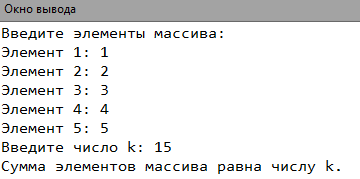
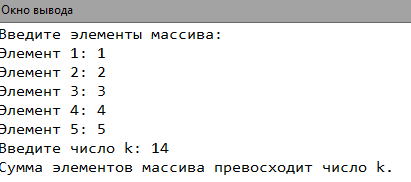
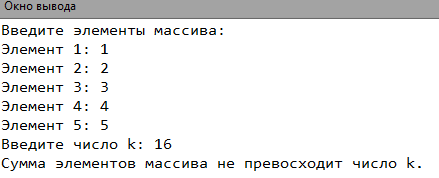
****

Рис. 3 – Результат выполнения программы №1

Задание 2.

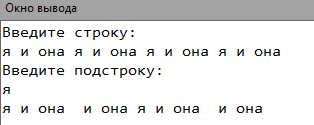
****

Рис. 4 – Результат выполнения программы №2

1. **Вывод.**

Данная работа прошла успешно. В работе были рассмотрены операции с массивами, построение схем алгоритмов, а также написание кода на языке программирования Pascal, вследствие ввода данных и проверка работоспособности программы.

В работе были написаны программы, определяющие превосходит ли сумма элементов массива число k, удаляющие из строки каждое второе вхождение заданной подстроки.

Трудности в работе возникли в построении схемы алгоритма и соединении связей в схеме алгоритма.

Таким образом, в данной работе были операции с массивами, построение схемы алгоритма, был описан каждой алгоритм, и проверка данной программы на работоспособность и анализ результат по ранее написанному коду.