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

Криницын Олег Сергеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2022

1. **Цель домашней контрольной работы:**

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

1. **Формулировка задания:**

Вариант 11.

1. Написать программу, выполняющую поиск такого минимального k, что сумма первых k элементов массива превосходит заданное число s.
2. Написать программу, определяющую является ли заданная строка подпоследовательностью исходной строки.
3. Все данные вводятся с клавиатуры. При выполнении второго пункта запрещается использовать стандартные функции для работы со строками (за исключением функции определения длины строки).
4. **Схема алгоритма с комментариями:**

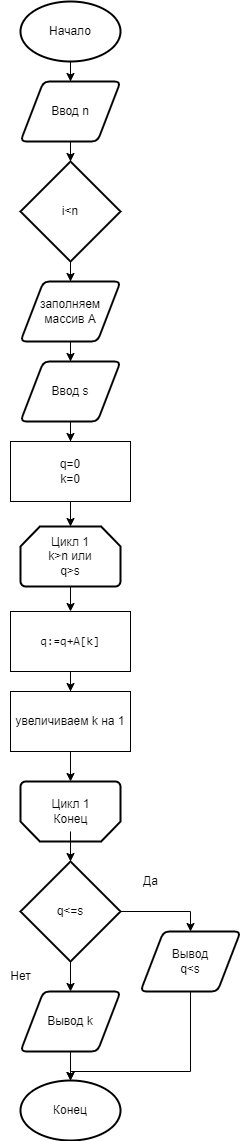
****

Рисунок 1 - Схема алгоритма задания 1

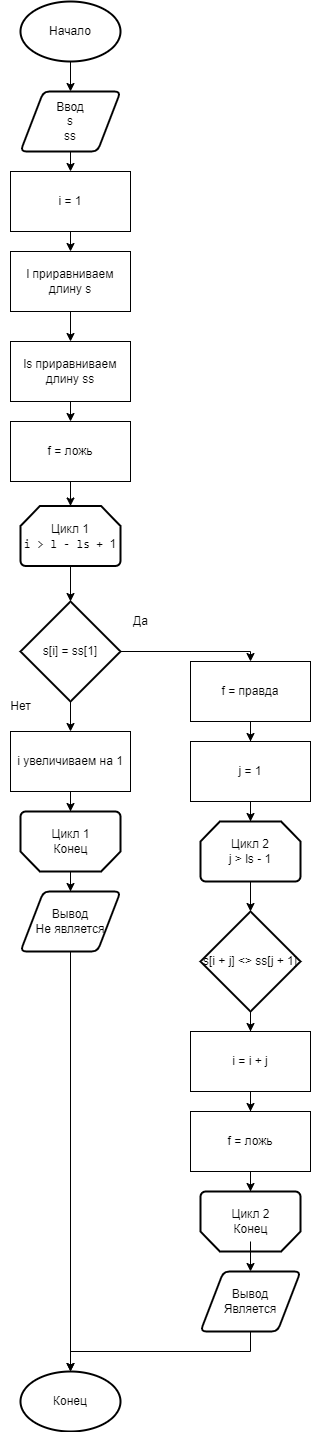


Рисунок 2 - Схема алгоритма задания 2

1. **Код программы**

**Задание 1**

**begin**

**var** A:**array of** integer;

**var** s: integer;

write('Размер = ');

**var** n:=readInteger;

SetLength(A, n);

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

**begin**

A[i]:= readInteger;

**end**;

writeln;

write('Рандом число: ');

readln(s);

**var** q:=0;

**var** k:=0;

**while** (k<n) **and** (q<=s) **do**

**begin**

inc(k);

q:=q+A[k];

**end**;

**if** q<=s **then** WriteLn('Сумма всех элементов = ',q,' < ',s)

**else**

WriteLn(k);

**end**.

**Задание 2**

**begin**

**var** s,ss: string;

**var** l,ls,i,j: integer;

**var** f: boolean;

write('Введите строку: ');

readln(s);

write('Введите подстроку: ');

readln(ss);

i := 1;

l := Length(s);

ls := Length(ss);

f := false;

**while** i <= l - ls + 1 **do**

**begin**

**if** s[i] = ss[1] **then**

**begin**

f := true;

**for** j := 1 **to** ls - 1 **do**

**if** s[i + j] <> ss[j + 1] **then**

**begin**

i := i + j;

f := false;

**break**

**end**;

**if** f **then**

**begin**

print('является');

**exit**;

**end**;

**end**;

i += 1;

**end**;

print('не является');

**end**.

1. **Результат выполнения программы**

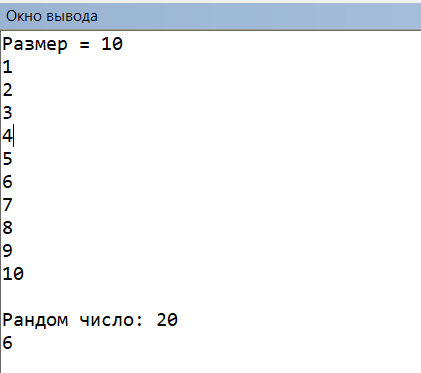
****

Рисунок 3 - Результат выполнения программы 1

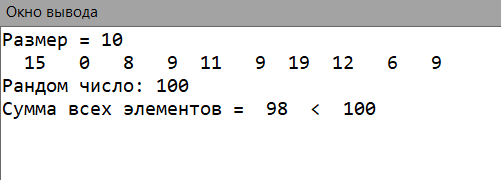


Рисунок 4 - Результат выполнения программы 1

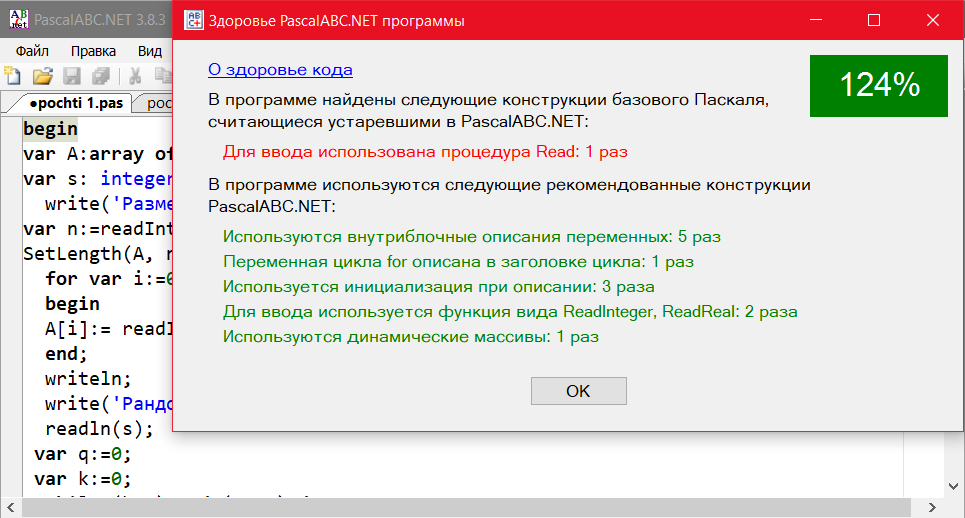


Рисунок 5 - Здоровье кода 1

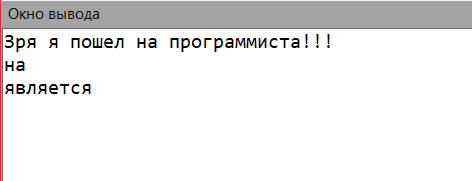


Рисунок 6 - Результат выполнения программы 2

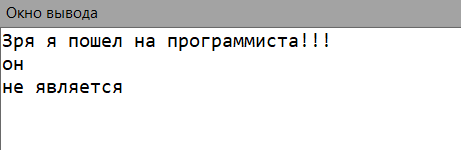


Рисунок 7 - Результат выполнения программы 2

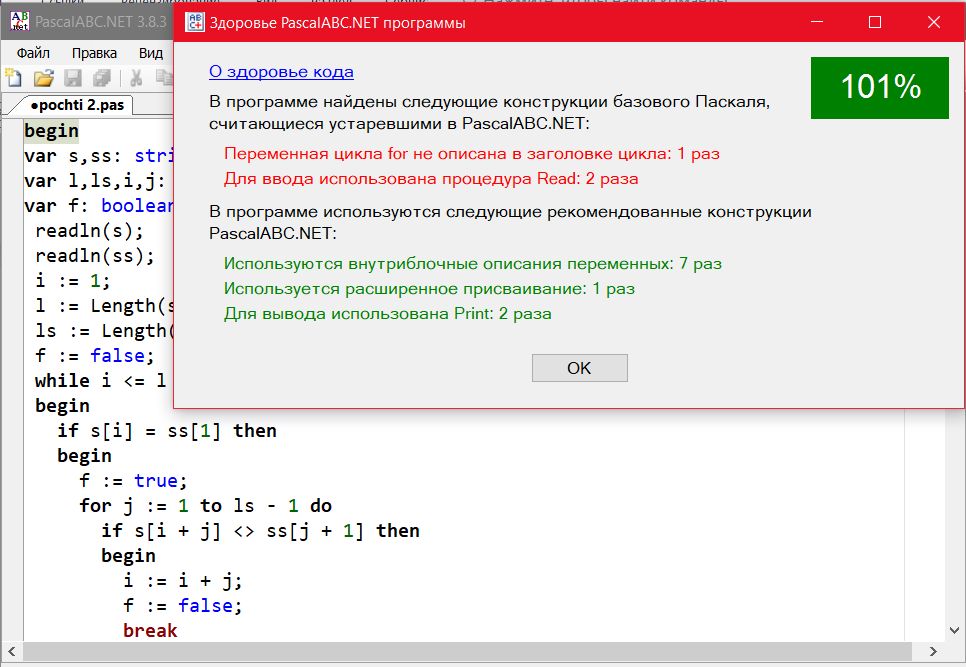


Рисунок 8 - Здоровье кода 2

1. **Вывод**

Нами была поставлена цель работы: получить базовые навыки работы с одномерными массивами, освоить принципы работы со строками как с частным случаем одномерных массивов. Работа была разделена на несколько этапов: цель, формулировка задания, создание схемы алгоритма, написание кода программы, результат выполнения программы, вывод. Схема алгоритма была создана с помощью сайта  [(https://app.diagrams.net/?src=about)](https://products.aspose.app/diagram/ru/flowchart). В схеме первого задания вводится размер массива и вручную заполняется любыми числами, используется один цикл и две проверки. В схеме второго задания вводятся две строки, используются два цикла и две проверки. Коды программ были написаны на PascalABC. Также важной задачей было Здоровье кода, у обеих программ он должен быть больше 100%. В первой программе Здоровье кода достигает 124%, во второй 101%. Итак, делая вывод, хочу скачать, что в ходе работы все цели были достигнуты.