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-204-52-00

Беляев Данил Игоревич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2022

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

Задание:

1. Написать программу, определяющую какие числа из интервала [1; к] не встречаются в

заданном массиве.

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).

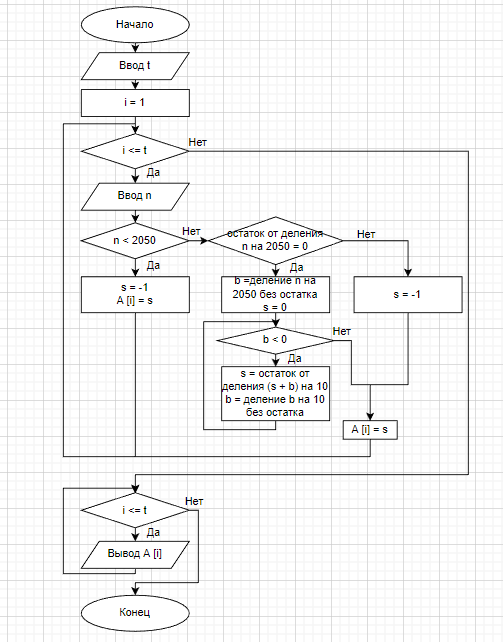


Рисунок 1 – схема алгоритма задания 1

**const**

Nmax = 10;

**type**

TArray = **array**[1..Nmax] **of** integer;

**procedure** Sort(**var** A:TArray);

**begin**

**var** i:integer;

**var** j:integer;

**var** tmp:integer;

**begin**

**for** i := high(A) **downto** low(A) + 1 **do**

**for** j := low(A) **to** pred(i) **do**

**if** A[j] > A[j + 1] **then**

**begin**

tmp := A[j];

A[j] := A[j + 1];

A[j + 1] := tmp;

**end**;

**end**;

**end**;

**procedure** ShowAbsence(**const** A: TArray; K: integer);

**begin**

**var** Previous: integer;

**var** i, j: integer;

**begin**

Previous := 0;

**for** i := low(A) **to** high(A) **do**

**if** (A[i] >= 1) **and** (A[i] <= K) **then**

**begin**

**for** j := Previous + 1 **to** pred(A[i]) **do**

Write(j: 4);

Previous := A[i];

**end**;

**for** j := Previous + 1 **to** K **do**

Write(j: 4);

Println;

**end**;

**end**;

**begin**

**var** A: TArray;

**var** K: integer;

**var** i: integer;

**begin**

K:=7;

randomize;

**for** i := low(A) **to** high(A) **do**

A[i] := -Nmax + random(2 \* Nmax);

**for** i := low(A) **to** high(A) **do**

Write(A[i]:4);

Println;

Sort(A);

**for** i := low(A) **to** high(A) **do**

Write(A[i]:4);

Println;

ShowAbsence(A,K);

**end**;

**end**.

Код программы задания 1

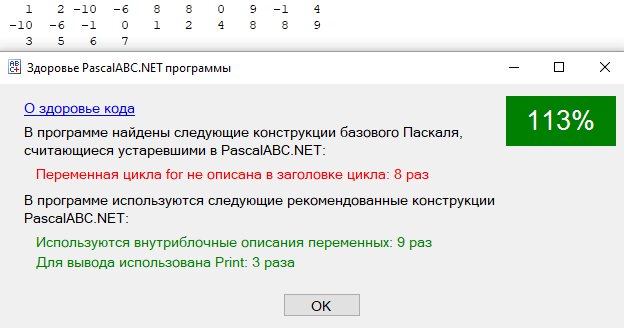


Рисунок 2 - результат программы задания 1

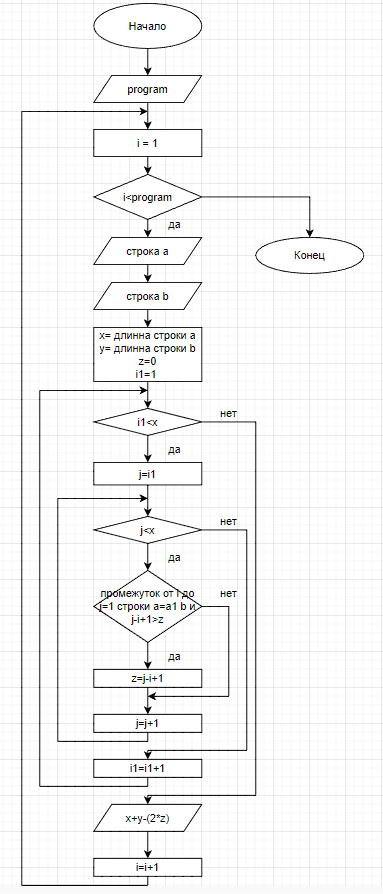
****

Рисунок 3 – схема алгоритма задания 2

**begin**

**var** s: string;

**var** ss: string;

**var** ls: integer;

**var** lss: integer;

**var** i: integer;

**var** j: integer;

**var** flag: boolean;

readln(s);

readln(ss);

i := 1;

ls := Length(s);

lss := Length(ss);

flag := false;

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

**begin**

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

**begin**

flag := true;

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

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

**begin**

i := i + j;

flag := false;

**break**

**end**;

**if** flag **then**

**begin**

print('является');

**exit**;

**end**;

**end**;

i += 1;

**end**;

print('не является');

**end**.

Код программы задания 2

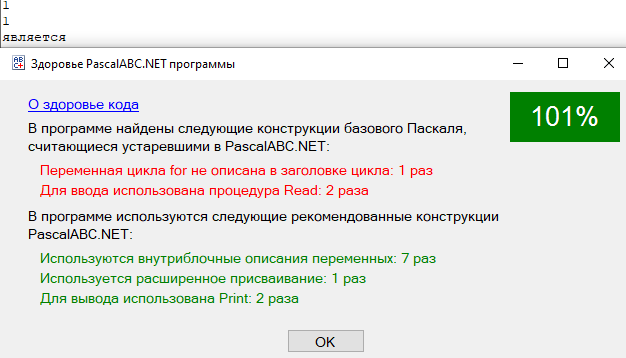


Рисунок 4 - результат программы задания 2

Вывод:

Задания были выполнены благодаря изучению одномерных массивов и принципов работы со строками в языке программирования Pascal.

В ходе выполнения задания были использованы структуры для работы с массивами (объявление массива определенной длины, обращение к определённому элементу массива по индексу в цикле, вывод заполненного массива) языка программирования Pascal.

Также были использованы структуры для работы со строками (объявление переменной со строковым типом данных, ввод строки и запись её содержания в переменную, обращение к определённому элементу строки по индексу в цикле, вычисление длины строки и запись значения длины в переменную) языка программирования Pascal.

Вместе со всем вышеперечисленным были использованы арифметические операторы языка Pascal для работы с переменными.  
Были применены операторы условия if и else, циклы с использованием оператора for, благодаря которым алгоритм выполнения программы был значительно упрощен.

С помощью всего вышеперечисленного задания были успешно выполнены.