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-208-52-00

ФИО: Лукьянов Илья Сергеевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

В отчете должны отображаться:

1. Цель работы

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

1. Формулировка заданий
2. Написать программу, определяющую превосходит ли сумма элементов массива число k.
3. Написать программу, выполняющую в строке замену заданной подстроки на инверсную ей.
4. Описание алгоритма №1

Алгоритм 1-ый:

1. Вводится массив.
2. Вводится индекс, до которого будет считаться сумма элементов массива.
3. Подсчитывается сумма.
4. Запускается цикл, который ищет в массиве такое число, которое было бы больше полученной ранее суммы.
5. Результат выводится на экран.

Алгоритм 2-ой:

1. Вводится основная строка.
2. Вводится подстрока строка.
3. Начинается цикл, который ищет каждое второе вхождение подстроки в основную строку и не позволяет её переписать если находит.
4. На экран выводится обновлённая основная строка.
5. Схема первого алгоритма

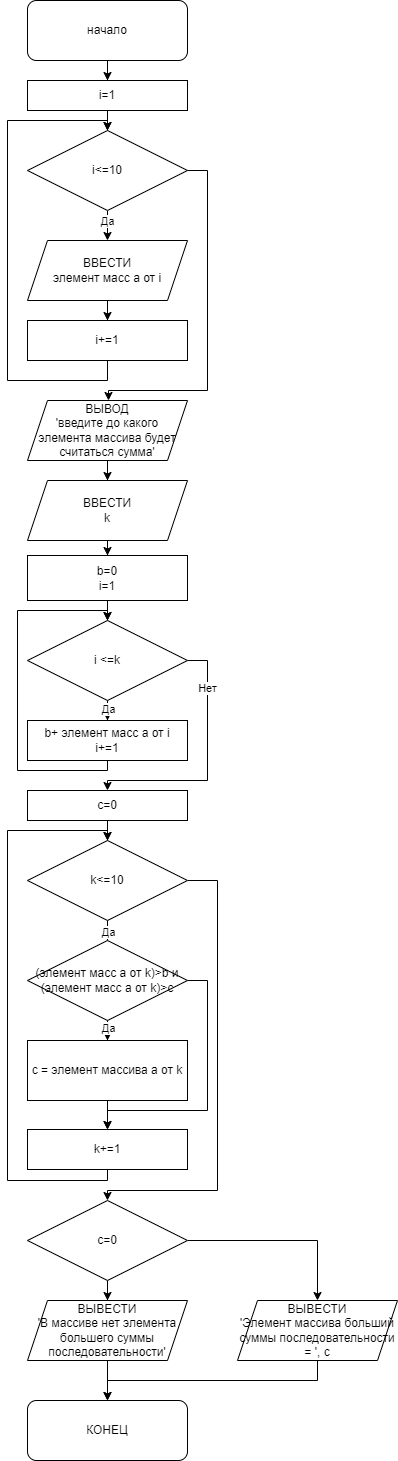


Рисунок 1 Алгоритм 1

1. Код программы для первой задачи
2. **var**
3. a:**array** [1..10] **of** integer;
4. i,k,b,c:integer;
6. **begin**
7. **for** i:=1 **to** 10 **do**
8. readln(a[i]);
9. writeln('введите до какого элемента массива будет считаться сумма');
10. readln(k);
11. b:=0;
12. **for** i:=1 **to** k **do**
13. b:=b+a[i];
14. c:=0;
15. **for** k:=k **to** 10 **do**
16. **if** (a[k]>b) **and** (a[k]>c) **then**
17. c:=a[k];
18. **if** c=0 **then**
19. writeln('В массиве нет элемента большего суммы последовательности')
20. **else**
21. writeln('Элемент массива больший суммы последовательности = ', c);
22. **end**.
23. Результат выполнения программы для первой задачи

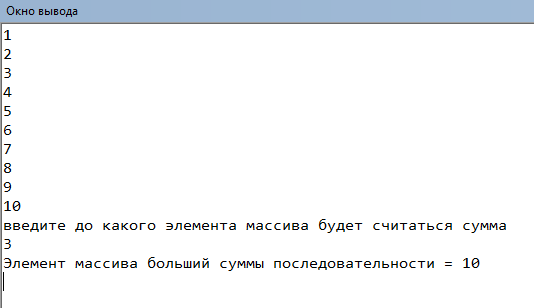


Рисунок 2 Результат алгоритма 1

1. Схема второго алгоритма

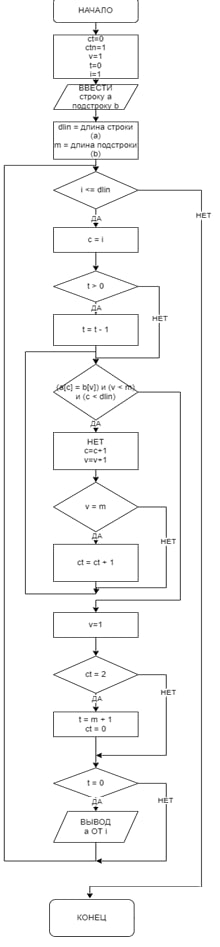


Рисунок 3 Алгоритм 2

1. Код программы для второй задачи
2. **Program** pr;
3. **var**
4. i,c,m,v,ct,t,dlin,ctn:integer;
5. a,b:string;
6. **begin**
7. ct:=0;
8. ctn:=1;
9. v:=1;
10. t:=0;
11. readln(a);
12. readln(b);
13. dlin:=Length(a);
14. m:=Length(b);
15. **for** i:=1 **to** dlin **do begin**
16. c:=i;
17. **if** t > 0 **then**
18. t:= t-1;
19. **while** (a[c] = b[v]) **and** (v < m) **and** (c < dlin) **do begin**
20. c:=c+1;
21. v:=v+1;
22. **if** v = m **then**
23. ct:=ct+1
24. **end**;
25. v:=1;
26. **if** ct=2 **then begin**
27. t:=m+1;
28. ct:=0;
29. **end**;
30. **if** t = 0 **then begin**
31. write(a[i]);
32. **end**;
33. **end**;
34. **end**.

1. Результат выполнения программы для второй задачи

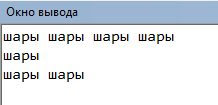


Рисунок 4 Результат алгоритма 1

1. Вывод

В данном отчете предоставлены данные и решения задач, которые были даны в домашней контрольной работе для написания и решения на языке программирования Pascal. Для выполнения были применены знания, которые давались на лекциях и практиках по дисциплине «Основы алгоритмизации и программирования. Так же, помогли базовые и более углубленные знания в математике, в частности, в алгебре.

Во время выполнения возникала проблема при выполнении задания с строкой. Решение было найдено после длительных попыток найти нужный алгоритм и перечитывания теории.