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-101-51-00

ФИО: Пономарёв Игнат Андреевич

Преподаватель:

Сергеева Елизавета Григорьевна

Киров

2023

В отчете должны отображаться:

1. Цель работы

Изучить базовую структуру организации программы и основные конструкции языка программирования Pascal.

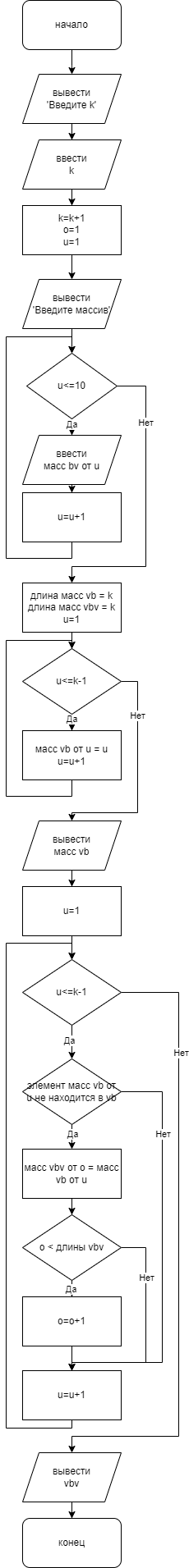
1. Формулировка заданий
2. Написать программу, определяющую какие числа из интервала [1;k] не встречаются в заданном массиве.
3. Написать программу, определяющую какое из слов встречается больше в заданной строке.
4. Описание алгоритма №1

Алгоритм 1-ый:

1. Определяется тип переменных.
2. Вводится k. .
3. Задаётся массив.
4. Заполняется массив от 1 до k с шагом 1.
5. Включается цикл, который сверяет числа в вписанном массиве и сверяет его со вторым массивом записывая все не встретившиеся цифры в отдельный массив.
6. На экран выводится массив с не встретившимися цифрами.

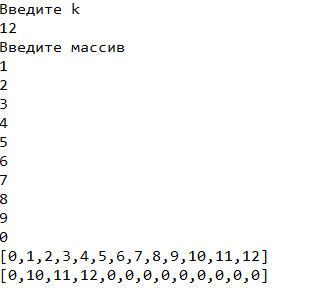
Алгоритм 2-ой:

1. Определяется тип переменных.
2. Вводится строка.
3. Включается цикл, который отделяет слова и сортирует каждое слово по отдельности во второй цикл.
4. С помощью цикла строка переводится в массив для удобства счета.
5. Запускается ещё один цикл, который берёт по одному слову из отсортированного массива и смотрит сколько раз оно встретилось в остальном.
6. Количество появлений каждого слова строки записывается в отдельный массив.
7. Включается третий цикл, который вычисляет, какое слово встретилось больше всего раз и выводит это слово на экран.
8. Схема первого алгоритма



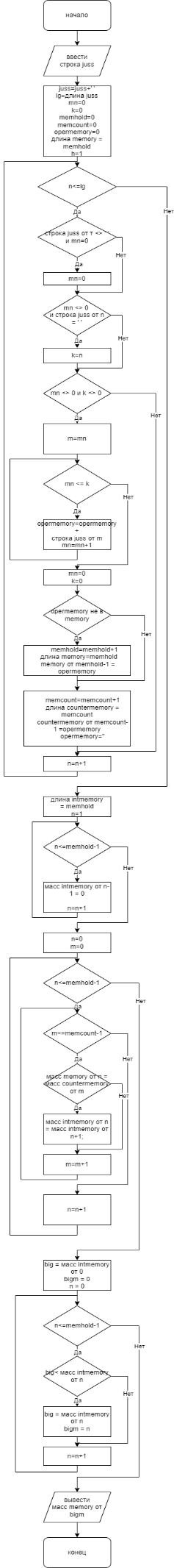
**Рисунок 1 Первый алгоритм**

1. Код программы для первой задачи
2. **var**
3. u,k,o:integer;
4. vb,vbv:**array of** integer;
5. bv:**array** [1..10] **of** integer;
7. **begin**
8. writeln('Введите k');
9. readln(k);
10. k:=k+1;
11. o:=1;
12. writeln('Введите массив');
13. **for** u:=1 **to** 10 **do**
14. readln(bv[u]);
15. SetLength(vb,k);
16. SetLength(vbv,k);
17. **for** u:=1 **to** k-1 **do begin**
18. vb[u]:=u;
19. **end**;
20. writeln(vb);
21. **for** u:=1 **to** k-1 **do begin**
22. **if** vb[u] **not in** bv **then begin**
23. vbv[o]:=vb[u];
24. **if** o<Length(vbv) **then**
25. o:=o+1;
26. **end**;
27. **end**;
28. writeln(vbv);
29. **end**.
30. Результат выполнения программы для первой задачи



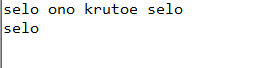
**Рисунок 2 Результат**

1. Схема второго алгоритма



**Рисунок 3 Второй алгоритм**

1. Код программы для второй задачи
2. **var**
3. juss,opermemory:string;
4. memory,countermemory:**array of** string;
5. intmemory:**array of** integer;
6. n,m,mn,k,lg,memhold,memcount,big,bigm:integer;
8. **begin**
9. readln(juss);
10. juss:=juss+' ';
11. lg:=Length(juss);
12. mn:=0;
13. k:=0;
14. memhold:=0;
15. memcount:=0;
16. opermemory:='';
17. SetLength(memory,memhold);
18. **for** n:=1 **to** lg **do begin**
19. **if** (juss[n]<>' ') **and** (mn=0) **then**
20. mn:=n;mol
21. **if** (mn<>0) **and** (juss[n]=' ') **then**
22. k:=n;
23. **if** (mn<>0) **and** (k<>0) **then begin**
24. **for** m:= mn **to** k **do begin**
25. opermemory:=opermemory+juss[m];
26. **end**;
27. mn:=0;
28. k:=0;
29. **if** opermemory **not in** memory **then begin**
30. memhold:=memhold+1;
31. SetLength(memory,memhold);
32. memory[memhold-1]:=opermemory;
33. **end**;
34. memcount:=memcount+1;
35. SetLength(countermemory,memcount);
36. countermemory[memcount-1]:=opermemory;
37. opermemory:='';
38. **end**;
39. **end**;
40. SetLength(intmemory,memhold);
41. **for** n:=1 **to** memhold-1 **do begin**
42. intmemory[n-1]:=0;
43. **end**;
44. **for** n:=0 **to** memhold-1 **do begin**
45. **for** m:=0 **to** memcount-1 **do begin**
46. **if** memory[n]=countermemory[m] **then begin**
47. intmemory[n]:=intmemory[n]+1;
48. **end**;
49. **end**;
50. **end**;
51. big:=intmemory[0];
52. bigm:=0;
53. **for** n:=0 **to** memhold-1 **do begin**
54. **if** big < intmemory[n] **then begin**
55. big:=intmemory[n];
56. bigm:=n;
57. **end**;
58. **end**;
59. writeln(memory[bigm]);
60. **end**.
61. Результат выполнения программы для второй задачи



**Рисунок 4 Результат**

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 решения домашней контрольной работы возникли большие трудности с решением второй задачи. Не хватало осознания итогового кода. Пришлось обратиться к коллегам и знакомым, которые смогли помочь решить задачу.