Министерство образования Республики Беларусь

Учреждение образования

БелорусскиЙ государственный университет

информатики и радиоэлектроники

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

|  |
| --- |
|  |
|  |
|  |

# оТЧЕТ

по лабораторной работе

на тему:

ПРОСТЫЕ ЦИКЛЫ

|  |  |  |
| --- | --- | --- |
| Выполнил  Студент гр. 951002 |  | А. А. Радько |
| Проверил |  | Асс. Е.Е. Фадеева |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Минск, 2019

1. Теоретические сведения по теме лабораторной работы

**Алгоритм** - система правил, четко описывающая последовательность действий, которые необходимо выполнить для решения задачи.

**Способы описания алгоритмов:**

1. запись на естественном языке (словесное описание)
2. изображение в виде схемы (графическое описание)
3. запись на алгоритмическом языке (составление программы)

**Критерии правильного алгоритма:**

1. **Дискретность** – значения величин в каждый следующий момент времени должны получаться по определенным правилам из значений величин, имевшихся в предшествующий момент времени.
2. **Определенность (детерминированность)** – каждое правило алгоритма должно быть однозначным. Значения величин, получаемых в какой-то момент времени, однозначно связаны со значениями величин, вычисленных ранее.
3. **Результативность (конечность)** – алгоритм должен приводить к решению задачи за конечное число шагов.
4. **Массовость** – алгоритм должен разрабатываться в общем виде так, чтобы его можно было применить для класса задач, различающихся лишь исходными данными.

**Цикл** – процесс, в котором отдельные участки вычислений выполняются многократно.

**Классификация циклов:**

В соответствии со взаимным расположением циклов в теле программы

или алгоритма различают следующие циклы:

**1) простые** – циклы, не содержащие внутри себя других циклов;

**2) сложные** – циклы, содержащие внутри себя другие циклы;

**а) вложенные** **(внутренние)** – циклы, входящие в состав других циклов (цикл в цикле);

**б) внешние** – циклы, не являющиеся составной частью других циклов, но

содержащие в своем составе внутренние циклы.

В зависимости от местоположения условия выполнения цикла различают

следующие циклы:

1) циклы с предусловием;

2) циклы с постусловием.

В соответствии с видом условия выполнения циклы делятся на следующие

виды:

1) циклы с параметром;

2) итерационные циклы.

* 1. Постановка задачи  
       
     Необходимо ввести два числа А и В. Проверить, если у них совпадающие цифры.
  2. Эскиз ожидаемого результата

Введите числа А и В

a

b

Есть совпадающие цифры

или

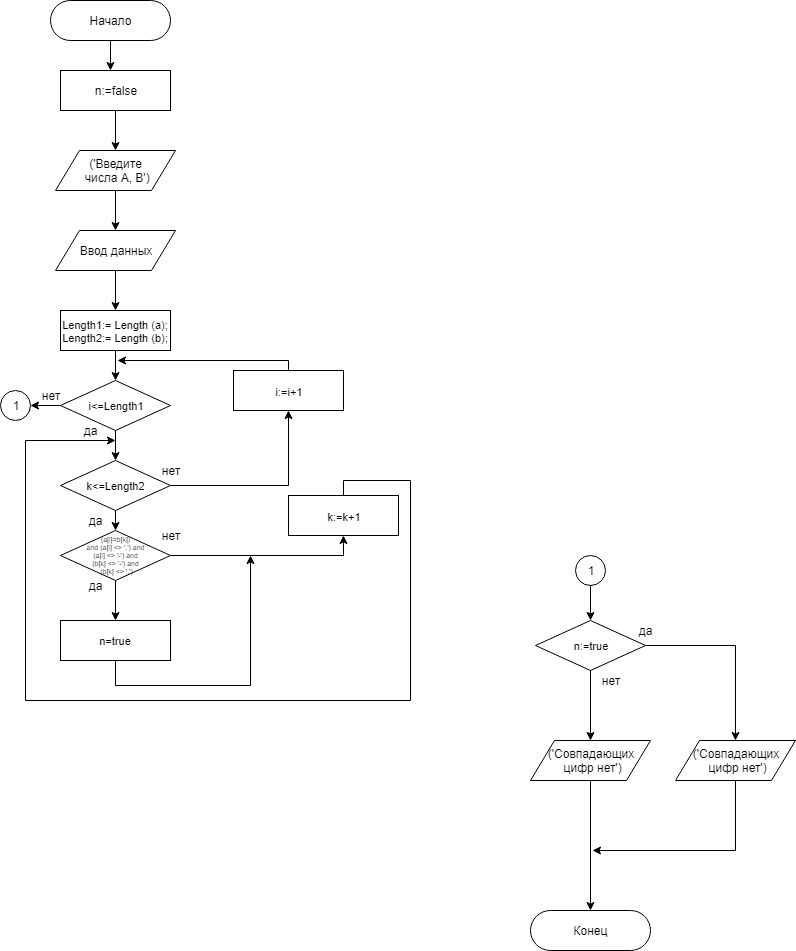
Совпадающих цифр нет

1. Выполнение
   1. Разработка алгоритма

Таблица 3.1 используемые идентификаторы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Имя  идентификатора | назначение | Тип идентификатора | Начальное значение | Закон изменения | Имя цикла, в котором происходит изменение переменной |
| a | Число А | string | Ввод с клавиатуры | - | - |
| b | Число B | string | Ввод с клавиатуры | - | - |
| i | Позиция в числе | integer | i:= 1 | i:= i+1 | Цикл А |
| k | Позиция в числе | integer | k:= 1 | k:= k+1 | Цикл В |
| n | Вспомогательная переменная | boolean | n:=false | - | - |
| Length1 | Длинна числа А | integer | - | - | - |
| Length2 | Длинна числа В | integer | - | - | - |

Рисунок 3.1 – Схема работы программы



3.2 Текст программы и его описание

program dop1;

{$APPTYPE CONSOLE}

{$R \*.res}

var

a: string[255];

b: string[255];

var i,k,Length1, Length2: integer;

n: boolean;

begin

n:=false;

writeln('Введите числа А и В');

readln(a);

readln(b);

Length1:= Length (a);

Length2:= Length (b);

for i := 1 to Length1 do // loop A

for k := 1 to Length2 do // loop B

begin

if (a[i]=b[k]) and (a[i] <> '.') and (a[i] <> '-') and (b[k] <> '-') and (b[k] <> '.')

then n:=true;

end;

if n then write('Есть совпадающие цифры')

else write('Совпадающих цифр нет');

end.

Текст программы

* 1. Тестирование и отладка программы

Таблица 3.2 Прохождение тестов программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Специфика тестирования | Номер теста | Вводимые данные | Ожидаемый результат | Полученный результат |
| Для отрицательных целых чисел, у которых совпадающих цифр нет | 1 | -55 -62 | Введите числа А и В  -55  -62  Совпадающих цифр нет | Тест пройден |
| Для отрицательных дробных чисел, у которых совпадающие цифры есть | 2 | -64.5 -74.3 | Введите числа А и В  -64  -74  Есть совпадающие цифры | Тест пройден |
| Для отрицательных дробных чисел, у которых совпадающих цифр нет | 3 | -2.2 -0.5 | Введите числа А и В  -2.2  -0.5  Совпадающих цифр нет | Тест пройден |
| Для целого отрицательного числа и дробного положительного чисел | 4 | -25 0.4 | Введите числа А и В  -25  0.4  Совпадающих цифр нет | Тест пройден |
| Для положительных целых чисел | 5 | 0 0 | Введите числа А и В  0  0  Есть совпадающие цифры | Тест пройден |

* 1. Итоговый текст программы

program dop1;

{$APPTYPE CONSOLE}

{$R \*.res}

var

a: string[255];

b: string[255];

var i,k,Length1, Length2: integer;

n: boolean;

begin

n:=false;

writeln('Введите числа А и В');

readln(a);

readln(b);

Length1:= Length (a);

Length2:= Length (b);

for i := 1 to Length1 do // loop A

for k := 1 to Length2 do // loop B

begin

if (a[i]=b[k]) and (a[i] <> '.') and (a[i] <> '-') and (b[k] <> '-') and (b[k] <> '.')

then n:=true;

end;

if n then write('Есть совпадающие цифры')

else write('Совпадающих цифр нет');

end.