БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра программного обеспечения информационных технологий

Факультет ЗВиДО

Специальность ПОИТ

Контрольная работа № 1

по дисциплине «Основы алгоритмизации и программирования»

часть 1

Вариант № 7

Выполнила: Н. Г. Карпеко

Договор № 696 от 16.05.2017г.

Минск 2018

**Тема задания**

**Циклический вычислительный процесс**. Составить алгоритм решения задачи. Нарисовать 3 варианта графического представления алгоритма:

а) в соответствии с ГОСТ 19.701-90;

б) по методу Дамке;

в) используя диаграмму Насси-Шнейдермана.

Алгоритмы изобразить средствами Word.

1. **Задание**

Найти числа, для которых перестановка последней цифры в начало увеличивает его в N раз.

1. **Схема алгоритма**

На рис. 1, 2, 3 приведены три способа графического представления алгоритма.

Да

Нет

Нет

Да

Нет

Да

Введите N

a := 10

a < 1000000

конец

c := a div 10

b := a mod 10

k := 1

d := a

Начало

d>=10

d := d div 10

k := k \* 10

s := c + b \* k

s / c = N

a := a + 1

Вывод a, s

Рис. 1. Схема алгоритма по ГОСТ 19. 701-90

конец

c := a div 10

b := a mod 10

k := 1

d := a

d := d div 10

k := k \* 10

While

a < 1000000

Начало

a := 10

Поиск чисел,

кратных N

Задание

кратности N

While

d>=10

s := c + b \* k

Нет

Да

s / c = N

a := a + 1

Вывод a, s

Рис. 2. Схема алгоритма, представленная методом Дамке

a < 1000000

Ввод значения кратности N

a := 10

b := a mod 10

Да

Нет

c := a div 10

k := 1

d := a

d >= 10

d := d div 10

k := k \* 10

s := c + b \* k

s := c + b \* k

вывод s, a

a := a + 1

Рис. 3. Диаграмма Насси-Шнейдермана

**3. Программа**

**program** Kratnost;

**var**

c, d, s, a: longint; {с- число на разряд меньшее, чем А}

{d- это А, чтобы не менять значение А и определить

разрядность числа А}

{a-Начальное число, до 1000000}

{s-Конечное число (искомое)}

N, b, k: integer; {N- кратность (мы задаем)}

{b-последняя цифра начального числа А}

{k - разрядность числа А}

**begin**

write('N= '); read(N);

a := 10; { -число должно быть минимум двузначным}

**while** a < 1000000 **do** {пока диапазон до 1 млн}

**begin**

b := a **mod** 10; {b - последняя цифра в начальном числе А}

c := a **div** 10; {с - на разряд меньшее число имеем}

k := 1; {к - разрядность числа А}

d := a; {чтобы узнать разрядность числа А}

**while** d >= 10 **do**

**begin**

d := d **div** 10; {разбили на 10ки для подсчета кол-ва разрядов. В конце

цикла получим d, равное числу < 10}

k := k \* 10; {k - разрядность (ск-ко циклов, столько раз умножится на 10)}

**end**;

s := c + b \* k; {новое число (переставили последнюю цифру в начало)}

**if** s / a = N **then** {условие нахождения чисел задачи}

**begin**

writeln(' a = ', a, ' s = ', s);

a := a + 1; {переходим к следующему числу}

**end**

**else** a := a + 1; {переходим к следующему числу}

**end**;

**end**.

**4. Тестирование программы**

Тест 1.

N= 1

a = 11 s = 11

a = 22 s = 22

a = 33 s = 33

a = 44 s = 44

a = 55 s = 55

a = 66 s = 66

a = 77 s = 77

a = 88 s = 88

a = 99 s = 99

a = 111 s = 111

a = 222 s = 222

a = 333 s = 333

…………………

a = 999 s = 999

a = 1111 s = 1111

a = 2222 s = 2222

…………………

a = 777777 s = 777777

a = 888888 s = 888888

a = 999999 s = 999999

Тест 4

N= 4

a = 102564 s = 410256

a = 128205 s = 512820

a = 153846 s = 615384

a = 179487 s = 717948

a = 205128 s = 820512

a = 230769 s = 923076

Тест 5

N= 5

a = 142857 s = 714285