

# Итерационные циклические вычислительные процессы с управлением по функции.

**Цель работы:** научиться использовать ИЦВП с управлением по функции.

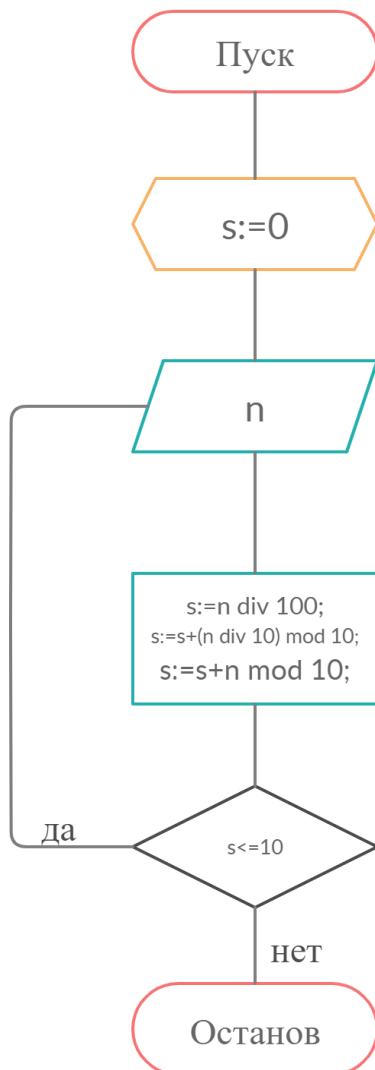
**Оборудование:** компьютер, PascalABC, Creately.

## Задание 1

**Постановка задачи:** С клавиатуры вводится трехзначное число, считается сумма его цифр. Если сумма цифр числа больше 10, то вводится следующее трехзначное число, если сумма меньше либо равна 10 – программа завершается.

**Математическая модель:**

**Блок схема:**



**Список идентификаторов:**

Переменная	Тип	Смысл
n	integer	Число, вводимое с клавиатуры
s	integer	Сумма цифр числа n

Код программы на PascalABC.NET:

```

program zd_1;
var n,s: integer;
begin
    repeat
        read(n);
        s:=n div 100;
        s:=s+(n div 10) mod 10;
        s:=s+n mod 10;
    until s<=10;
end.

```

Результат программы:

```

program zd_1;
var n,s: integer;
begin
    repeat
        read(n);
        s:=n div 100;
        s:=s+(n div 10) mod 10;
        s:=s+n mod 10;
    until s<=10;
end.

```

<

Окно вывода

560  
123

Анализ:

Программа работает корректно, реализована с помощью пост-проверки, так как с пред-проверкой реализовать программу не получится.

## **Задание 2 1**

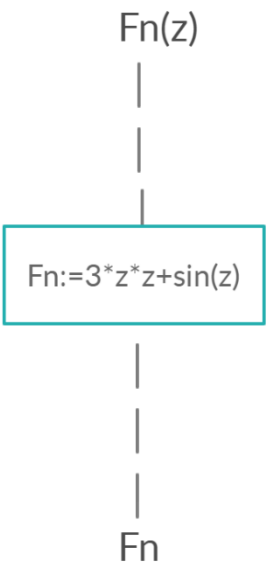
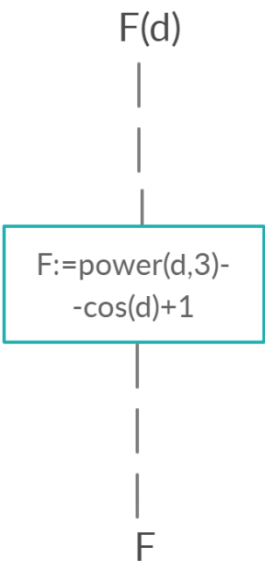
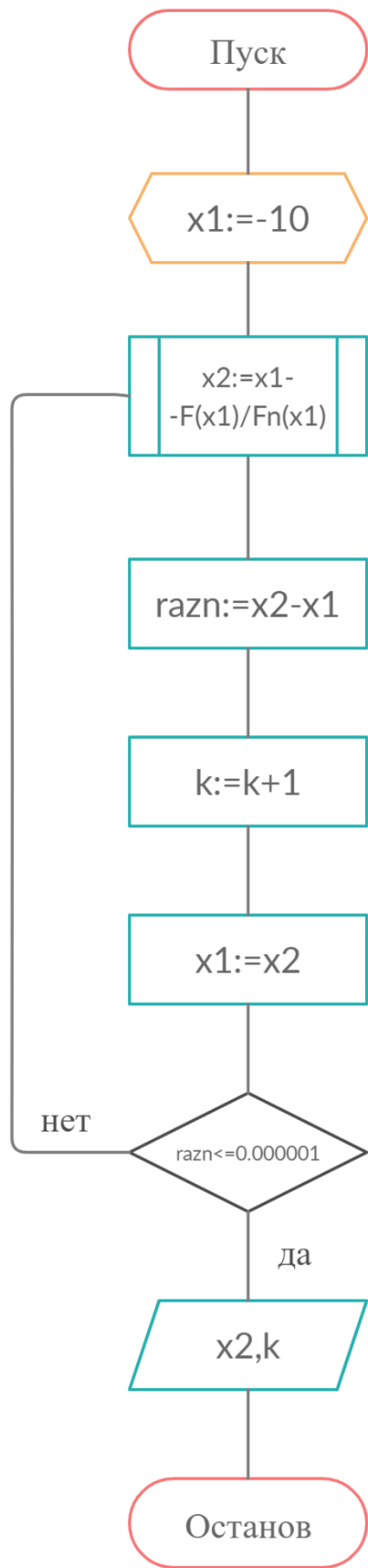
**Постановка задачи:** Решить нелинейное уравнение методом Ньютона.

**Математическая модель:**

$$f(x) := x^3 - \cos(x) + 1 \text{ на отрезке от } -10 \text{ до } 10 \text{ с точностью } 10^{-6}$$

---

Блок схема:



## Список идентификаторов:

Переменная	Тип	Смысл
X1	real	Аргумент
X2	real	Следующий аргумент
razn	real	Разница аргументов
k	integer	счётчик
d	Real	Переменная в функции
z	Real	Переменная в функции
F	real	Функция уравнения от x
Fn	real	Производная от уравнения

## Код программы на PascalABC.NET:

```
program zd_2;
var x1,x2,razn: real;
var k: integer;
function F(d:real): real;
begin
    F:=power(d,3)-cos(d)+1;
end;

function Fn (z:real): real;
begin
    Fn:=3*z*z+sin(z);
end;

begin
    x1:=-10;
    repeat
        x2:=x1-F(x1)/Fn(x1);
        razn:=x2-x1;
        k:=k+1;
        x1:=x2;
    until razn<=power(10,-6);
    writeln(x2:8:7);
    writeln(k);
end.
```

## Результат программы:

```

function F(d:real): real;
begin
  F:=power(d,3)-cos(d)+1;
end;

function Fn (z:real): real;
begin
  Fn:=3*z*z+sin(z);
end;

begin
  x1:=-10;
  repeat
    x2:=x1-F(x1)/Fn(x1);
    razn:=x2-x1;
    k:=k+1;
    x1:=x2;
  until razn<=power(10,-6);
  writeln(x2:8:7);
  writeln(k):

```

Окно вывода

```

0.1648390
0.0880121
0.0315818
0.0042826
0.0000762
0.0000000
-0.4900726
13

```

### Анализ:

Программа нашла корень уравнения спустя 13 повторений цикла

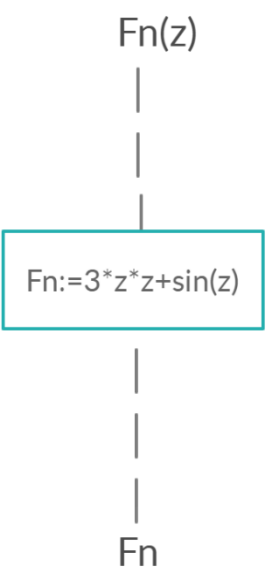
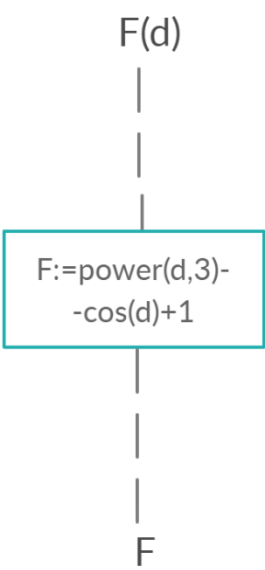
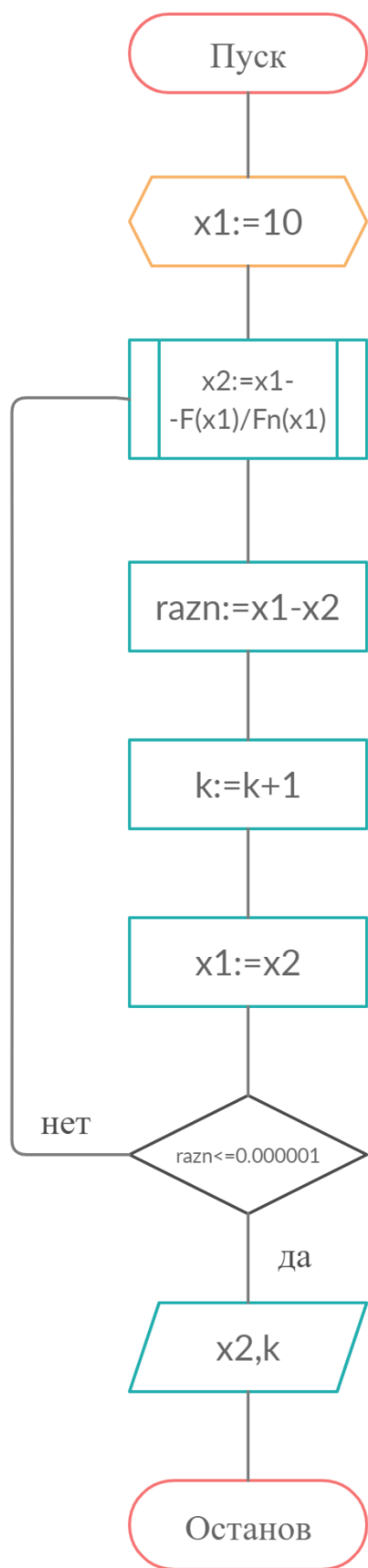
## Задание 2 2

**Постановка задачи:** Решить нелинейное уравнение методом Ньютона.

**Математическая модель:**

$f(x) := x^3 - \cos(x) + 1$  на отрезке от -10 до 10 с точностью  $10^{-6}$

**Блок схема:**



**Список идентификаторов:**

Переменная	Тип	Смысл
X1	real	Аргумент
X2	real	Следующий аргумент
razn	real	Разница аргументов
k	integer	счётчик
d	Real	Переменная в функции
z	Real	Переменная в функции
F	real	Функция уравнения от x
Fn	real	Производная от уравнения

### Код программы на PascalABC.NET:

```

function F(d:real): real;
begin
    F:=power(d,3)-cos(d)+1;
end;

function Fn (z:real): real;
begin
    Fn:=3*z*z+sin(z);
end;

begin
    x1:=10;
    repeat
        x2:=x1-F(x1)/Fn(x1);
        razn:=x1-x2;
        k:=k+1;
        x1:=x2;
    until razn<=power(10,-6);
    writeln(x2:8:7);
    writeln(k);
end.

```

### Результат программы:



```

function F(d:real): real;
begin
    F:=power(d,3)-cos(d)+1;
end;

function Fn (z:real): real;
begin
    Fn:=3*z*z+sin(z);
end;

begin
    x1:=10;
    repeat
        x2:=x1-F(x1)/Fn(x1);
        razn:=x1-x2;
        k:=k+1;
        x1:=x2;
    until razn<=power(10,-6);
    writeln(x2:8:7);
    writeln(k):

```

<

Окно вывода

0.0000008

27

### Анализ:

Программа нашла второй корень уравнения за 27 повторений цикла

### Общий вывод:

Я научился использовать ИЦВП с управлением по функции.