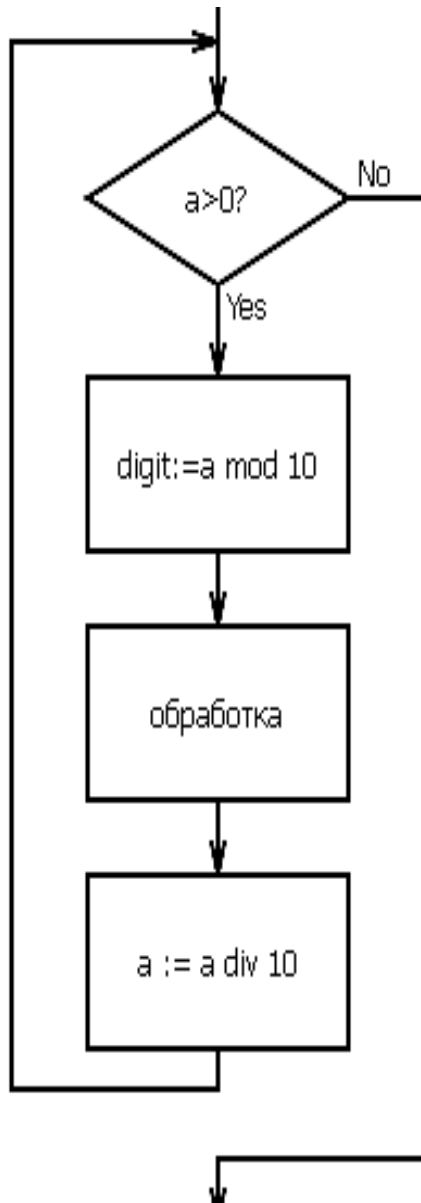


Задание 20

Тип 1 ВЫДЕЛЕНИЕ ЦИФР ИЗ ЧИСЛА



1. пока цифры в числе есть
2. выделить последнюю цифру
3. обработать её
4. удалить цифру из числа

$a \bmod b$ – остаток от деления
числа a на число b

$a \div b$ – целая часть от деления
числа a на число b

Общий подход к решению – «действуй с конца»

1. Что выводится? Чему равны переменные вывода??
2. Определить тип алгоритма
3. Составить математическую модель.
4. Нахождение искомого числа.

Пример 1. При каком **наибольшем** n программа выведет на экран сначала 3, а потом 2?

```
var n, a, b, c:
integer;
begin
  readln(n);
  a := 0; b := 0;
  while n > 0 do
  begin
    a := a+1;
    c := n mod 10;
    b := b+c;
    n := n div 10;
  end;
  writeln(a);
  write(b);
end.
```

1. Определяем конечные значения переменных:
 $a = 3, b = 2$.
 2. Определяем тип алгоритма:
выделение цифр из числа.
 a – считает итерации \rightarrow считает разряды
исходного числа,
 b – результат обработки цифр числа,
 $n \text{ div } 10$ – десятичная система счисления.
 3. Математическая модель:
 $a=3 \rightarrow n$ – трёхзначное число.
 $b=b+c=2 \rightarrow$ сумма цифр числа n равна 2.
 4. Вывод: возможные цифры искомого числа -
(2,0,0) или (1,1,0)
- Т.к. нам надо **наибольшее**,
то получаем число 200.

Ответ: 200.

Пример 2. Укажите **наибольшее из таких чисел x , при вводе которых алгоритм печатает сначала 3, а потом 8.**

```
var x, L, M:
integer;
begin
  readln(x);
  L:=0; M:=0;
  while x > 0 do
begin
  L:= L + 1;
  if x mod 2 = 0
then
M:= M + x mod 10;
x:= x div 10;
end;
writeln(L);
write(M);
end.
```

1. Определяем конечные значения переменных:

$$L = 3, M = 8.$$

2. Определяем тип алгоритма:

выделение цифр из числа.

L – считает итерации \rightarrow считает разряды исходного числа,

M – результат обработки цифр числа,

$x \text{ div } 10$ – десятичная система счисления.

3. Математическая модель:

$L=3 \rightarrow x$ – трёхзначное число.

$M=8$ – сумма четных цифр числа x .

4. Вывод: т.к. нам надо *наибольшее*, то это число, состоящее из цифр (8,9,9).

Наибольшее такое число 998.

Ответ: 998.

Пример 3. Укажите **наименьшее число x , при вводе которого алгоритм печатает сначала 3, а потом 6.**

```
var x, L, M:
integer;
begin
  readln(x);
  L := 0;
  M := 0;
  while x > 0 do
  begin
    M := M + 1;
    if x mod 2 <> 0
then
      L := L + 1;
      x := x div 2;
    end;
    writeln(L);
    writeln(M);
  end.
```

1. Определяем конечные значения переменных:

$$L = 3, M = 6.$$

2. Определяем тип алгоритма:

выделение цифр из числа.

M – считает итерации \rightarrow считает разряды
исходного числа,

L – результат обработки цифр числа,

$x \text{ div } 2$ – двоичная система счисления.

3. Математическая модель:

$M=6 \rightarrow$ двоичная запись числа x содержит
шесть разрядов.

$L=L+1$, если $x \bmod 2 \neq 0 \rightarrow$ количество
нечетных цифр числа x .

4. Вывод: x – наименьшее число, в двоичной
записи которого всего шесть разрядов, из них -
три нечетные цифры.

$$(1,1,1,0,0,0) \rightarrow x = 100011_2 = 35_{10}.$$

Ответ: 35.

Пример 4. Укажите **наибольшее число x , при вводе которого алгоритм печатает сначала 3, а потом 120.**

1. Определяем конечные значения переменных:

$$L = 3, M = 120.$$

2. Определяем тип алгоритма:

выделение цифр из числа.

L – считает итерации \rightarrow считает разряды
исходного числа,

M – результат обработки цифр числа,

$x \text{ div } 8$ – восьмеричная система счисления.

3. Математическая модель:

$L=3 \rightarrow$ восьмеричная запись

числа x содержит три разряда.

$M=120 \rightarrow$ произведение цифр

восьмеричного числа x .

4. Вывод: возможные цифры искомого числа -
(6,5,4) или

Т.к. нам надо **наибольшее**, то получаем 654_8 .

$$x = 654_8 = 428_{10}.$$

Ответ: 428.

```
var x, L, M:
longint;
begin
  readln(x);
  L:=0; M:=1;
  while x > 0 do
begin
  L:=L+1;
  M:= M*(x mod 8);
  x:= x div 8;
end;
  writeln(L);
  write(M);
end.
```

Пример 5. Укажите **наибольшее число x , при вводе которого алгоритм печатает сначала 3, а потом 1.**

```
var x, L, M: integer;
begin
  readln(x);
  L:=0; M:=0;
  while x > 0 do
  begin
    L:=L+1;
    if M < (x mod 10)
  then begin
      M:=x mod 10;
    end;
    x:= x div 10;
  end;
  writeln(L);
  write(M);
end.
```

1. Определяем конечные значения переменных:

$$L = 3, M = 1.$$

2. Определяем тип алгоритма:

выделение цифр из числа.

L – считает итерации \rightarrow считает разряды
исходного числа,

M – наибольшая? (наименьшая?) цифра числа,
 $x \text{ div } 10$ – десятичная система счисления.

3. Математическая модель:

$L=3 \rightarrow$ число x трехзначное

$M=1 \rightarrow$ наибольшая? наименьшая? цифра

$M=1 \rightarrow$ наибольшая цифра числа x

4. Вывод: наибольшее трехзначное число, в котором наибольшая цифра 1, это число 111.

Ответ: 111.

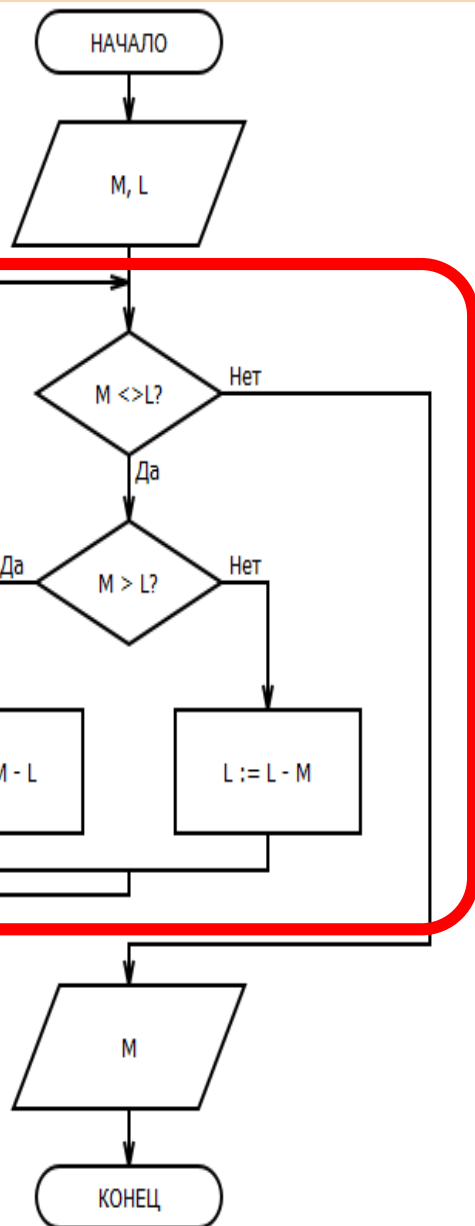
Пример 6. Укажите **наименьшее пятизначное число , при вводе которого алгоритм печатает сначала 4, а потом 2.**

```
var x, y, a, b:
longint;
begin
    a := 0;
    b := 0;
    readln(x);
    while x > 0 do
begin
    y := x mod 10;
    if y > 3 then a:=a+1;
    if y < 8 then b:=b+1;
    x := x div 10
end;
    writeln(a);
    writeln(b)
end.
```

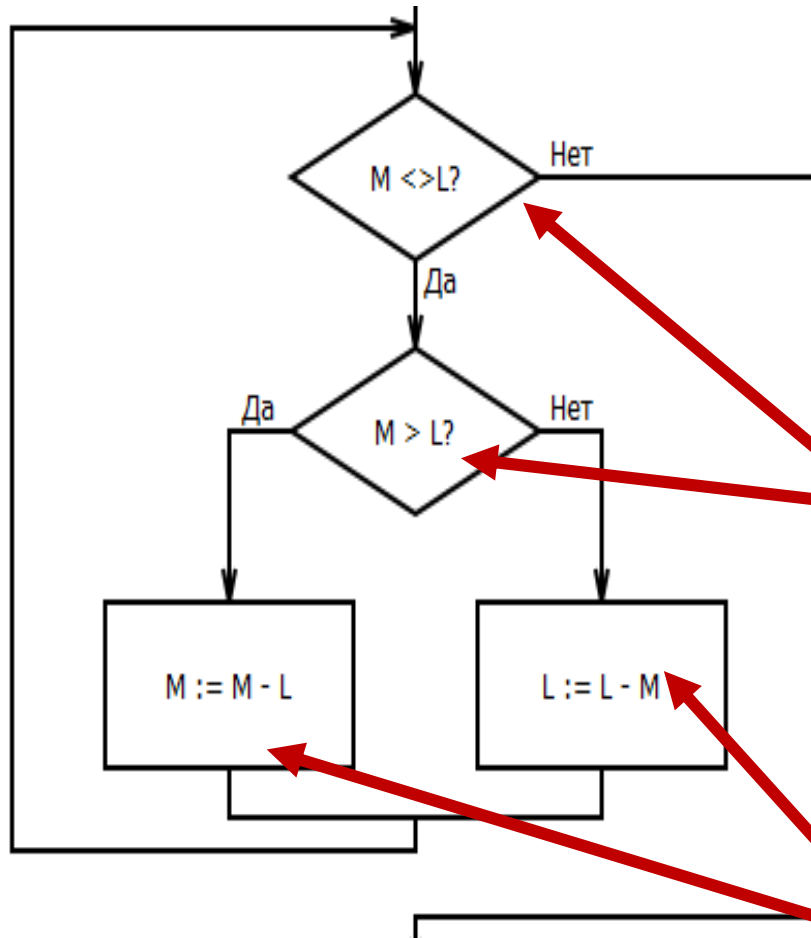
1. Определяем конечные значения переменных: $a = 4, b = 2$.
 2. Определяем тип алгоритма:
выделение цифр из числа.
 a – счетчик \rightarrow считает количество цифр числа, больших 3,
 b – счетчик \rightarrow считает количество цифр числа, больших 8.
 3. Математическая модель:
 $a=4 \rightarrow$ в числе x 4 цифры, большие 3
 $b=2 \rightarrow$ в числе x 2 цифры, меньше 8
 4. Вывод:
а) так как число пятизначное, есть $4 + 2 - 5 =$ одна цифра, которая больше 3 и меньше 8 одновременно; она должна быть **минимальной**, поэтому эта цифра **4**
б) для того, чтобы число было минимальным, еще одна цифра должна быть **минимальной** и меньшей 3 – это старшая цифра 1
б) три цифры минимальные из цифр, большие или равные 8, это три цифры 8.
- Ответ: 14888.**

Нахождение НОД (наибольший общий делитель) двух чисел.

- пока числа M и L не равны
- сравнить числа
- большее заменить разницей большего и меньшего



КАК ОПРЕДЕЛИТЬ??



Признаки:

- 1. В цикле всего две переменные!**
- 2. В цикле два условия: условие цикла и поиск большего!**
- 3. Наличие разностей!**

Реализация на Паскале алгоритма Евклида

```
function NOD (a, b: integer): integer;  
begin  
    while a <> b do  
        if a > b  
            then a := a - b  
            else b := b - a;  
    NOD := a;  
end;
```

Пример 1. Получив на вход число x , этот алгоритм печатает число M . Известно, что $x > 100$. Укажите **наименьшее** такое число x , при вводе которого алгоритм печатает 26.

```
var x, L, M:integer;  
begin  
  readln(x);  
  L := x;  
  M := 65;  
  if L mod 2 = 0  
then  
    M := 52;  
  while L <> M do  
    if L > M then  
      L := L - M  
    else  
      M := M - L;  
  writeln(M);  
end.
```

1. Определяем конечные значения переменных $M=26$.

2. Тип алгоритма - алгоритм Евклида.

3. $26 = \text{НОД}(x, M)$.

4. Разбираемся с M .

а) если x (оно же L) – нечетное,
то $M := 65$

б) если x (оно же L) – четное,
то $M := 52$

5. Находим x

а) $\text{НОД}(x, 65) = 26$ – не может быть, так как 65 не делится на 26

б) $\text{НОД}(x, 52) = 26$, $x > 100$

$x = 104$ – не подходит, так как 104 кратно 52 и $\text{НОД}(104, 52) = 104$

$x = 104 + 26 = 130$ – подходит

Ответ: 130

Пример 2. Получив на вход число x , этот алгоритм печатает число M . Известно, что $x > 50$. Укажите **наименьшее** такое число x , при вводе которого алгоритм печатает 14.

```
var x, L, M:integer;  
begin  
  readln(x);  
  L := x;  
  M := 35;  
  if L mod 2 = 0  
then  
    M := 28;  
  while L <> M do  
    if L > M then  
      L := L - M  
    else  
      M := M - L;  
  writeln(M);  
end.
```

1. Определяем конечные значения переменных $M=14$.

2. Тип алгоритма - алгоритм Евклида.

3. $14 = \text{НОД}(x, M)$.

4. Разбираемся с M .

а) если x (оно же L) – нечетное,
то $M := 35$

б) если x (оно же L) – четное,
то $M := 28$

5. Находим x

а) $\text{НОД}(x, 35) = 14$ – не может быть, так как 35 не делится на 14

б) $\text{НОД}(x, 28) = \underline{14}$, $x > 50$

$x=56$ – не подходит, так как 56 кратно 28 и $\text{НОД}(56, 28) = \underline{28}$

$x=56+14=70$ – подходит

Ответ: 70

Пример 3. Получив на вход число x , этот алгоритм печатает число M . Известно, что $x > 100$. Укажите **наименьшее** такое число x , при вводе которого алгоритм печатает 11.

```
var x, L, M:integer;  
begin  
  readln(x);  
  L := x-21;  
  M := x+12;  
  while L <> M do  
    if L > M then  
      L := L - M  
    else  
      M := M - L;  
  writeln(M);  
end.
```

1. Определяем конечные значения переменных $M=11$.
2. Тип алгоритма - алгоритм Евклида.
3. $11=\text{НОД}(L, M)$. L и M делятся на 11.
4. Находим L , *чтобы x при этом был наименьшим* и больше 100, и при этом L должно делиться на 11.
5. Подбираем. Это число 88.
L=77: $x=77+21=98 < 100$ - не подходит
L=88: $x=88+21=109$
L=88, $X=109$, $M=109+12=121$

Ответ: 109

Пример 4. Получив на вход число x , этот алгоритм печатает число M . Известно, что $x > 100$. Укажите **наименьшее** такое число x , при вводе которого алгоритм печатает 35.

```
var x, L, M: integer;  
begin  
  readln(x);  
  L := x-15;  
  M := x+20;  
  while L <> M do  
    if L > M then  
      L := L - M  
    else  
      M := M - L;  
  writeln(M);  
end.
```

1. Определяем конечные значения переменных $M=35$.
2. Тип алгоритма - алгоритм Евклида.
3. $35 = \text{НОД}(L, M)$. L и M делятся на 35.
4. Находим L , *чтобы x при этом был наименьшим* и больше 100, и при этом L должно делиться на 35.
5. Подбираем. Это число 105.
 $L=105$: $x=105+15=120$ - подходит
 $L=105, X=120, M=120+20=140$

Ответ: 120