

Простейшие алгоритмы

1. Сумма и произведение

```
var s := 0;
for var i:=1 to n do
  s += xi
var p := 1;
for var i:=1 to n do
  p *= xi
```

2. Арифметическая прогрессия

```
var x := a;
loop n do
  begin
    Print(x);
    x += d;
  end;
for var i:=0 to n-1 do
  Print(a + i*d);
```

3. $n! = n*(n-2)*(n-4)*... * 2$ (или 1)

```
var p := 1;
var x := n;
while x >= 2 do
  begin
    p *= x;
    x -= 2;
  end;
```

4. Сколько нечетных среди n введенных

```
var c := 0;
loop n do
  begin
    var x := ReadInteger;
    if x mod 2 <> 0 then
      c += 1;
  end;
```

4а. С помощью ReadSeqInteger и foreach

```
var sq := ReadSeqInteger(n);
var c := 0;
foreach var x in sq do
  if x mod 2 <> 0 then
    c += 1;
```

5. Защита от неверного ввода. Сокращенное вычисление логических выражений

```
var i: integer;
Print('Введите i (>0):');
while not TryRead(i) or not (i>0) do
  Print('Неверный ввод. Повторите:');
```

```
var i: integer;
repeat
  Print('Введите i (>0):');
until TryRead(i) and (i>0);
```

6. Табулирование функции f(x) на [a,b] в точках, разбивающих [a,b] на n частей

```
Assert(N>0);
var h := (b-a)/n;
var x := a;
for var i:=0 to n do
  begin
    WritelnFormat('{0,5:f2} {1,9:f4}',x,f(x));
    x += h;
  end;
```

6а. Решение, использующее while. Погрешность округления и вычислительная погрешность

```
var h := (b-a)/N;
var x := a;
while x <= b+h/2 do
  begin
    WritelnFormat('{0,5:f2} {1,9:f4}',x,f(x));
    x += h;
  end;
```

6б. Partition и последовательность точек

```
foreach var x in Partition(a,b,n) do
  Println(x, f(x));
```

7. Вывод n первых чисел Фибоначчи

```
Assert(n>1);
var (a,b) := (1,1);
Print(a,b);
loop n-2 do
  begin
    var c := a + b;
    Print(c);
    a := b;
    b := c;
  end;
```

7а. С помощью множественного присваивания

```
var (a,b) := (1,1);
Print(a,b);
loop n-2 do
  begin
    (a,b) := (b,a+b);
    Print(b);
  end;
```

8. Найти НОД(a,b), используя алгоритм Евклида: $\text{НОД}(a,b) = \text{НОД}(b, a \bmod b)$; $\text{НОД}(a,0) = a$

```
var (a,b) := ReadInteger2;
repeat
  var c := a mod b;
  a := b;
  b := c;
until b=0;
Print(a);
```

8а. С помощью множественного присваивания

```
var (a,b) := ReadInteger2;
repeat
  (a,b) := (b, a mod b);
until b=0;
```

9. Найти сумму цифр целого положительного числа n

```
var m := ReadInteger;
var s := 0;
while m>0 do
  begin
    s += m mod 10;
    m := m div 10;
  end;
```

Максимумы и минимумы

10. Найти max из введенных чисел

```
var x := ReadReal;
var max := x;
loop n do
  begin
    x := ReadReal;
    if max < x then
      max := x;
  end;
```

10а. С помощью ReadSeqReal и foreach

```
var max := real.MinValue;
foreach var x in ReadSeqReal(n) do
  if max < x then
    max := x;
```

11. Найти минимальный x, удовлетворяющий условию p(x)

```
var min := real.MaxValue;
loop n do
  begin
    var x := ReadReal;
    if (x < min) and p(x) then
      min := x;
  end;
if min = real.MaxValue then
  Println('Нет удовл. условию');
```

Суммирование рядов и нахождение предела последовательности

12. Вычислить $\sum_{i=1}^n \frac{a^i}{i!}$

```
Read(a, n);
var x := a;
var s := x;
for var i := 2 to n do
  begin
    x *= a / i;
    s += x;
  end;
```

13. Вычислить $\lim_{n \rightarrow \infty} a_n = \sqrt{x}$.

$$a_1 = 1, a_{n+1} = \frac{1}{2} \left(a_n + \frac{x}{a_n} \right)$$

```
Assert(x>=0);
var x := ReadReal;
var eps := 1e-10;
var a := 1.0;
var b := real.MaxValue;
while abs(b - a) >= eps do
  begin
    b := a;
    a := (a + x/a) / 2;
  end;
Print(a, Sqrt(x));
```

13а. С помощью множественного :=

```
var (a,b) := (1.0, real.MaxValue);
while abs(b - a) >= eps do
  (b,a) := (a, (a + x / a) / 2);
```

Поиск значения

14. Есть ли среди введенных число k?

```
var Exists := False;
loop n do
  begin
    var x := ReadInteger;
    if x = k then
      Exists := True;
  end;
```

14а. То же с использованием break

```
var Exists := False;
loop n do
  begin
    var x := ReadInteger;
    if x = k then
      break;
  end;
end;
```

14б. То же с использованием while

```
var Exists := False;
var i := 1;
while (i<=n) and not Exists do
  begin
    var x := ReadInteger;
    i += 1;
    if x = k then
      Exists := True;
  end;
```

Другие алгоритмы

15. Является ли число n>1 простым?

```
var IsPrime := True;
// for var i:=2 to n-1 do
for var i:=2 to Round(Sqrt(n)) do
  if n mod i = 0 then
    begin
      IsPrime := False;
      break;
    end;
```

16. Разложение числа на простые множители

```
Assert(x>=2);
var i := 2;
repeat
  if x mod i = 0 then
    begin
      Print(i);
      x := x div i;
    end
  else i += 1;
until x = 1;
```

17. Дана непрерывная на [a,b] функция f(x), имеющая на [a,b] ровно один корень (f(a)*f(b)<=0). Найти его методом половинного деления

```
Assert(b>a);
var (fa,fb) := (f(a), f(b));
Assert(fa*fb <= 0);
while (b-a) > eps do
  begin
    var x := (b + a)/2;
    var fx := f(x);
    if fa*fx <= 0 then
      b := x;
    else (a,fa) := (x,fx);
  end;
Println((b + a)/2);
```