а) РБНФ;

б) диаграмм Вирта;

в) формальных грамматик.

**цифра**

1

0

5

4

9

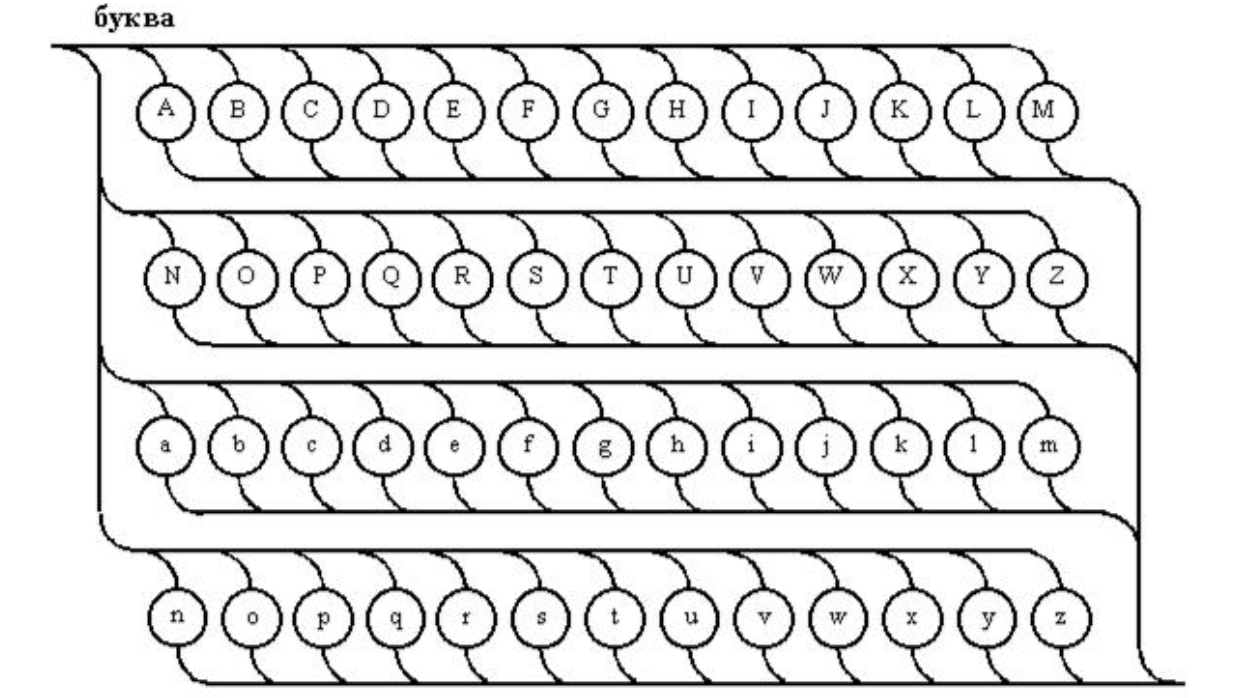
6

2

7

3

8



буква

цифра

буква

**идентификатор**

целое

**действительное:**

порядок

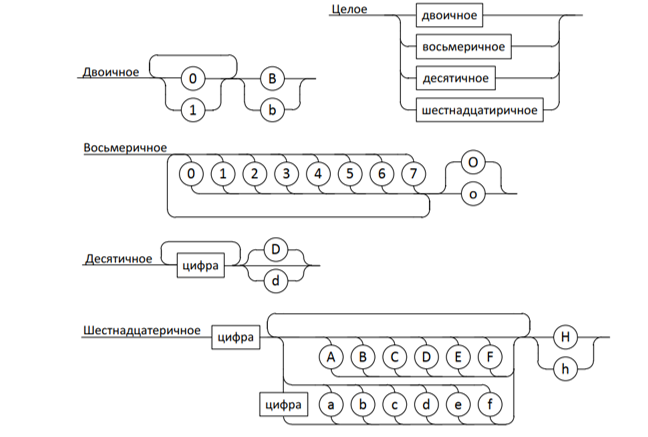
.,

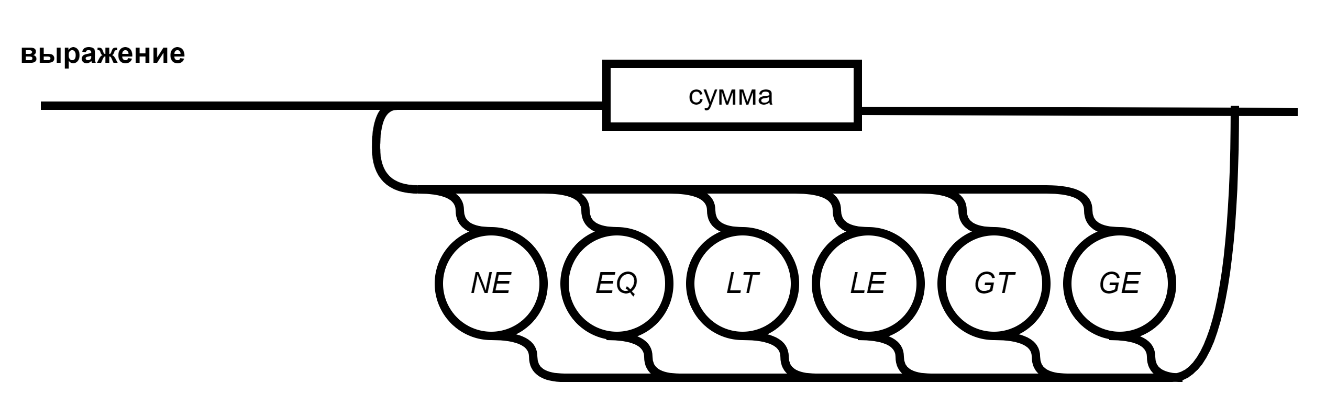
Числовая строка

Числовая строка

Числовая строка

Числовая строка





**сумма**

произведение

+

-

or

**произведение**

множитель

\*

/

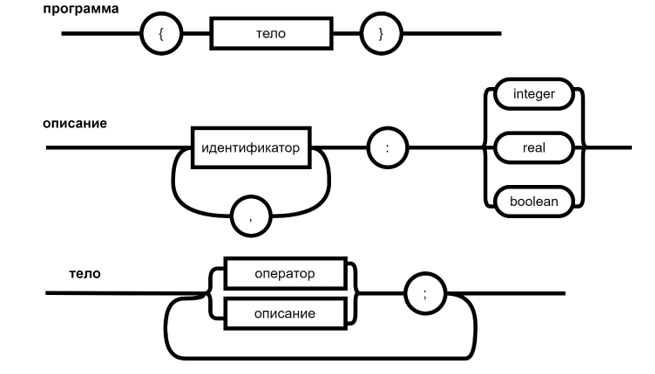
and

%

!

**тип**

$



**оператор**

присваивания

условный

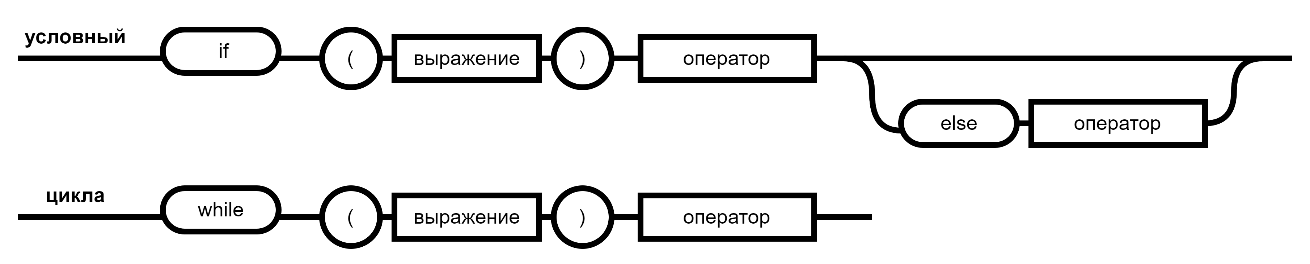
цикла

составной

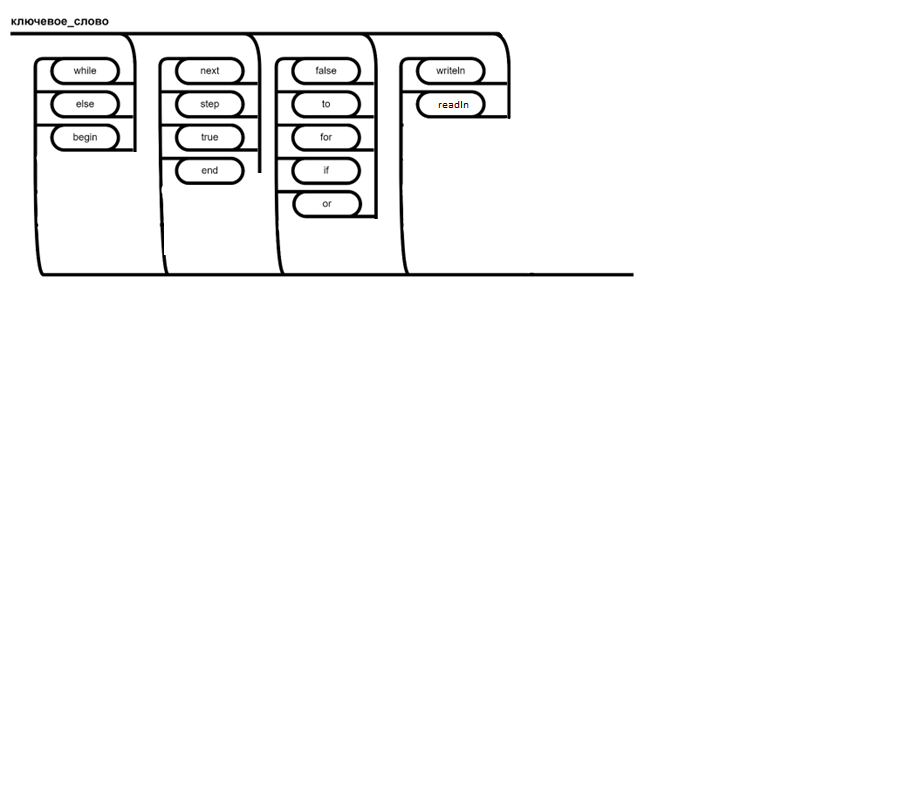
ввода

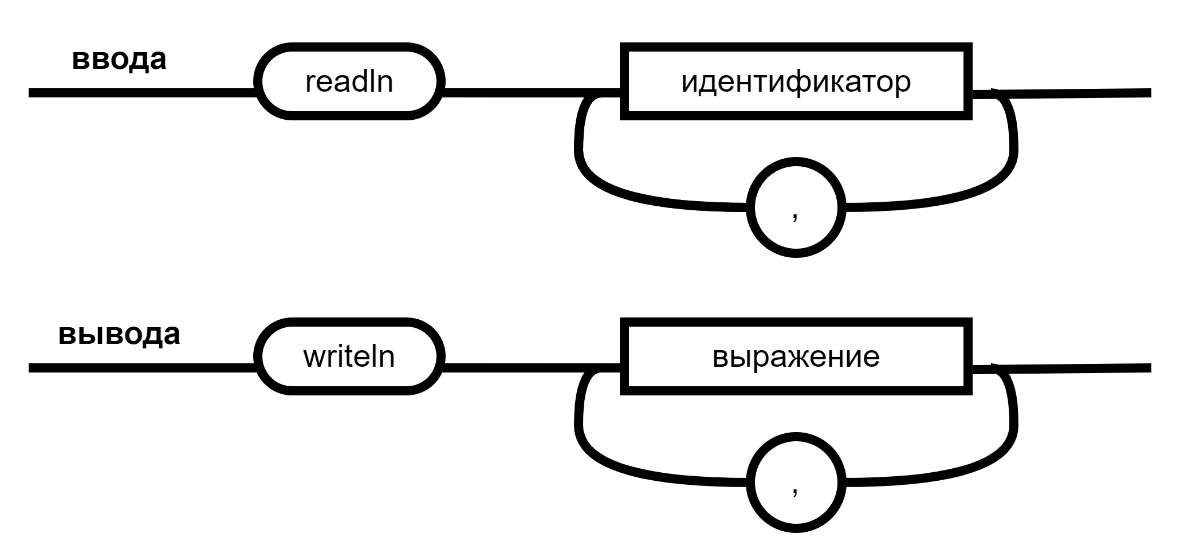
вывода











true

false

**логическая\_константа**

**тело**

begin

end

оператор

;

<буква>::= *A* | *B* | *C* | *D* | *E* | *F* | *G* | *H* | *I* | *J* | *K* | *L* | *M* | *N* | *O* | *P* | *Q* | *R* | *S* | *T* | *U* | *V* | *W* | *X* | *Y* | *Z* | *a* | *b* | *c* | *d* | *e* | *f* | *g* | *h* | *i* | *j* | *k* | *l* | *m* | *n* | *o* | *p* | *q* | *r* | *s* | *t* | *u* | *v* | *w* | *x* | *y* | *z*

<цифра> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

<идентификатор> ::= <буква> { <буква> | <цифра> }

<число> ::= < целое >| <действительное>

<программа> ::= *program var <описание> begin <оператор> {; <оператор>} end.*

<описание> ::= *dim* <идентификатор> {, <идентификатор> } <тип>

<оператор> ::= <присваивания> | <условный> | <цикла> | <составной> |

<ввода> | <вывода>

<присваивания> ::= <идентификатор> := <выражение>

<условный> ::= *if* «(»<выражение> «)» <оператор> *else* <оператор>

<цикла> ::= *while* «(»<выражение> «)» <оператор>

<фиксированного\_цикла>::= for <присваивания> to <выражение> [step <выражение>] <оператор> next

<составной>:: = *begin* <оператор> { ; <оператор> } *end*

<ввода>:: = *readln*(идентификатор)

<вывода>:: = *writeln*(выражение)

<выражение>:: = <сумма> | <сумма> (операции группы отношений ) <сумма>

<сумма> ::= <произведение> { (+ | - | and) <произведение>}

<произведение>:: = <множитель> { (\*| / | and)<множитель>}

<множитель>:: = <идентификатор> | <число> | <логическая\_константа> | not <множитель> | (<выражение>)

<логическая\_константа>:: = *true* | *false*

1

<ключевое\_слово> ::= 0*program* | 1*var* | 2*begin* | 3*end* | 4*readln* | 5*writeln* | 6if | 7*then* | 8*else* | 9*while* | 10*true* | 11*false |12 or | 13not*

2

<разделитель> ::= 1/\* |2 \*/ |3 , |4 ; |5 : |6 := |7 . |8 ( |9 ) |10 ~ |11+|12-|13<|14>|15<=|16>=|17<>|18\*| 19/

*Program* → program var *Descr* begin *Body* end.

*Body* → *Operat* | *Body; Operat*

*Descr* → dim *ID1* ! | dim *ID1* $ | dim *ID1* %

*ID1* → *ID | ID1, ID*

*ID* → *Char | ID Char | ID Digit1*

*Operat* → begin *Operat1* end | *ID* := *Expr* | if (*Expr*) *Operat* else *Operat* | if (*Expr*) *Operat* | while (*Expr*) *Operat* | for *ID* := *Expr* to *Expr Operat* next | for *ID* := *Expr* to *Expr* step *Expr Operat* next | readln *ID1* | writeln *Expr1*

*Expr1*→ *Expr | Expr1, Expr*

*Expr* → *Sum | Expr* <> *Sum | Expr* = *Sum | Expr* < *Sum | Expr* > *Sum | Expr* <= *Sum | Expr* >= *Sum*

*Sum* → *Mult | Sum* + *Mult | Sum* - *Mult | Sum* or *Mult*

*Mult* → *Fact | Mult* \* *Fact | Mult* / *Fact | Mult* and *Fact*

*Fact* → *ID | Num | LogConst |* not *Fact | (Expr)*

*LogConst* → true | false

*Num* → *Int | Real*

*Int* → *Bin | Oct | Dec | Hex*

*Bin* → *Bin1* B *| Bin1* b *| Bin1 Bin*

*Bin1* → 0 | 1

*Oct* → *Oct1* O *| Oct1* o *| Oct1 Oct*

*Oct1* → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7

*Dec* → *Digit1* D | *Digit1* d | *Digit1*

*Hex* → *Hex1* H *| Hex1* h

*Hex* → *Digit | Hex1 Digit | Hex1 Ch*

*Ch* → *a | b | c | d | e | f | A | B | C | D | E | F*

*Real* → *Digit1 Por | Digit1.Digit1.Por | .Digit1 Por | .Digit1 | Digit1.Digit1*

*Por* → E *+ Digit1 |* e *+ Digit1 |* E *- Digit1 |* e *- Digit1 |* E  *Digit1 |* e  *Digit1*

*Char* → *a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z*

*Digit1* → *Digit | Digit1 Digit*

*Digit* → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

{/\*Среднее арифметическое чисел, введенных с клавиатуры\*/

program var dim k, n, sum, itog %;

begin

readln n;

sum:= 0;

i:= 1;

while (i <= n)

begin

readln k;

sum:=sum + k;

i:= i + 1;

end

itog:= sum / n;

writeln itog;

end.

--------------------------------------------------------------------

{/\*Найти максимальное из 3х чисел\*/

program var dim a, b, c %;

begin readln a, b, c;

if (a > b) max:=a else max:= b;

if (c > max) max:= c;

writeln max;

end.

-------------------------------------------------------------------

{/\*Факториал числа\*/

program var dim a, i, rez %;

begin rez:= 1;

readln a;

for i:= 1 to a rez:= rez \* i next

writeln rez;

end.

-----------------------------------------------------------------

{/\*Проверка: может ли существовать треугольник\*/

program var dim a, b, c %;

otvet $;

begin readln a, b, c;

if (a >= (b + c)) otvet:= false

else if (b >= (a + с)) otvet:= false

else if (c >= (a + b)) otvet:= false

else otvet:= true;

writeln otvet;

end.

-----------------------------------------------------------

{/\*Возведение в степень\*/

program var dim a, i %;

begin

readln a, i;

for i:= 1 to i a=a\*a next writeln a;

end.