**Формы Бэкуса-Наура (БНФ)**

Метаязык, предложенный Бэкусом и Науром, использует следующие обозначения:

* символ «::=» отделяет левую часть правила от правой (читается: «определяется как»);
* нетерминалы обозначаются произвольной символьной строкой, заключенной в угловые скобки «<» и «>»;
* терминалы - это символы, используемые в описываемом языке;
* правило может определять порождение нескольких альтернативных цепочек, отделяемых друг от друга символом вертикальной черты «|» (читается: «или»).

**Пример**  Определение понятия «идентификатор» с использованием БНФ имеет вид:

<идентификатор> ::= <буква> | <идентификатор> <буква>

| <идентификатор> <цифра>

<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*

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

**Расширенные формы Бэкуса-Наура (РБНФ)**

Для повышения удобства и компактности описаний, в РБНФ вводятся следующие дополнительные конструкции (метасимволы):

* квадратные скобки «[» и «]» означают, что заключенная в них синтаксическая конструкция может отсутствовать;
* фигурные скобки «{» и «}» означают повторение заключенной в них синтаксической конструкции ноль или более раз;
* сочетание фигурных скобок и косой черты «{/» и «/}» используется для обозначения повторения один и более раз;
* круглые скобки «(» и «)» используются для ограничения альтернативных конструкций.

**Пример** В соответствии с данными правилами синтаксис модельного языка *М* будет выглядеть следующим образом:

<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*

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

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

<число> ::= {/< цифра> /}

<ключевое\_слово> ::= *program* | *var* | *begin* | *end* | *int* | *bool* | *read* | *write* | *if* |

*then* | *else* | *while* | *do* | *true* | *false*

<разделитель> ::= ( | ) | , | ; | : | := | . | { | } |+ | - | \* | / | ∨ | ∧ | ¬ | = | < | >

<программа> ::= *program* <описание> ; <тело>.

<описание> ::= *var* <идентификатор> {, <идентификатор>}: (*int* | *bool*)

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

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

<ввода> | <вывода>

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

<условный> ::= *if <*выражение> *then* <оператор> *else* <оператор>

<цикла> ::= *while <*выражение> *do* <оператор>

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

<ввода>:: = *read*(<идентификатор>)

<вывода>:: = *write*(<выражение>)

<выражение>:: = <сумма> | <сумма> ( = | < | >) <сумма>

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

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

<множитель>:: = <идентификатор> | <число> | <логическая\_константа> |

¬ <множитель> | (<выражение>)

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

**Пример** Программа на модельном языке *М*, вычисляющая среднее арифметическое чисел, введенных с клавиатуры.

*program*

*var k*, *n*, *sum*: *int*;

*begin*

*read*(*n*);

*sum*:= *0*;

*i*:=1;

*while i*<=*n do*

*begin*

*read*(*k*);

*sum*:=*sum*+*k*;

*k*:=*k*+1

*end*;

*write*(*sum*/*n*)

*end*.