|  |  |  |
| --- | --- | --- |
| **Identifiers** | | |
| id | *ID* | identifiers |
| **KeyWords** | | |
| package | *PACKAGE* | Keyword of package |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Function** | | |
|  |  |  |
|  |  |  |
| **BasicTypes (KeyWords)** | | |
|  |  |  |
|  |  |  |
|  |  |  |
| **Constants** | | |
|  |  |  |
|  |  |  |
| **Operators** | | |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| **Separators** | | |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Subgrammar “Programm Unit”**

G0 = <T,N, $TRANSLATED,R>

N = { $TRANSLATED }

T = { PACKAGE, ID, SCOLON, TRANSLATEDDEC }

R = {

1. $TRANSLATED ➝ PACKAGE ID SCOLON TRANSLATEDDEC
2. $TRANSLATED ➝ PACKAGE ID SCOLON

}

Отношения предшествования:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | id | package | scolon | translatedec | $ |
| id |  |  | [=.] |  |  |
| package | [=.] |  |  |  |  |
| scolon |  |  |  | [=.] | [.>] |
| translatedec |  |  |  |  | [.>] |
| # |  | [<.] |  |  |  |

Таблица переноса :

|  |  |
| --- | --- |
|  | € |
| # [package, id, scolon, translatedec] | 0, TRANSLATED |
| # [package, id, scolon] | 1, TRANSLATED |

Таблица свертки:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | id | package | scolon | translatedec | $ |
| [id, C] |  |  | S |  |  |
| [package, C] | S |  |  |  |  |
| [scolon, C] |  |  |  | S | R |
| [translatedec, C] |  |  |  |  | R |
| [#] |  | S |  |  |  |
| [#, TRANSLATED] |  | S |  |  | A |

**Subgrammar “Translate Unit”**

G1 = <T,N, $ TRANSLATEDDEC,R>

N = { $TRANSLATEDDEC, $IMPORT }

T = { CLASS, MODIFIER, ID, SCOLON, OB, CB, **CLASSBODY** }

R = {

1. $TRANSLATEDDEC ➝ $IMPORT MODIFIER CLASS ID OB **CLASSBODY** CB
2. $TRANSLATEDDEC ➝ $IMPORT MODIFIER CLASS ID OB CB
3. $TRANSLATEDDEC ➝ MODIFIER CLASS ID OB **CLASSBODY** CB
4. $TRANSLATEDDEC ➝ MODIFIER CLASS ID OB CB
5. $IMPORT ➝IMPORT ID SCOLON $IMPORT
6. $IMPORT ➝IMPORT ID SCOLON

}

Отношения предшествования:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | cb | class | classbody | id | import | modifier | ob | scolon | $ |
| cb |  |  |  |  |  |  |  |  | [.>] |
| class |  |  |  | [=.] |  |  |  |  |  |
| classbody | [=.] |  |  |  |  |  |  |  |  |
| id |  |  |  |  |  |  | [=.] | [=.] |  |
| import |  |  |  | [=.] |  |  |  |  |  |
| modifier |  | [=.] |  |  |  |  |  |  |  |
| ob | [=.] |  | [=.] |  |  |  |  |  |  |
| scolon |  |  |  |  | [<.] | [.>] |  |  |  |
| # |  |  |  |  | [<.] | [<.] |  |  |  |

Таблица переноса :

|  |  |
| --- | --- |
|  | € |
| # [TRANSLATEDDEC, modifier, class, id, ob, classbody , cb] | 0, TRANSLATEDDEC |
| # [TRANSLATEDDEC, modifier, class, id, ob, cb] | 1, TRANSLATEDDEC |
| # [modifier, class, id, ob, classbody , cb] | 2, TRANSLATEDDEC |
| # [modifier, class, id, ob, cb] | 3, TRANSLATEDDEC |
| scolon [import, id, scolon, TRANSLATEDDEC] | 4, TRANSLATEDDEC |
| # [import, id, scolon, TRANSLATEDDEC] | 4, TRANSLATEDDEC |
| scolon [import, id, scolon] | 5, TRANSLATEDDEC |
| # [import, id, scolon] | 5, TRANSLATEDDEC |

Таблица свертки:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | cb | class | classbody | id | import | modifier | ob | scolon | $ |
| [cb, C] |  |  |  |  |  |  |  |  | R |
| [class, C] |  |  |  | S |  |  |  |  |  |
| [classbody , C] | S |  |  |  |  |  |  |  |  |
| [id, C] |  |  |  |  |  |  | S | S |  |
| [import, C] |  |  |  | S |  |  |  |  |  |
| [modifier, C] |  | S |  |  |  |  |  |  |  |
| [ob, C] | S |  | S |  |  |  |  |  |  |
| [scolon, C] |  |  |  |  | S | R |  |  |  |
| [#] |  |  |  |  | S | S |  |  |  |
| [#, TRANSLATEDDEC] |  |  |  |  | S | S |  |  | A |

**Subgrammar “Class Body”**

G2 = <T,N, $ CLASSBODY,R>

N = { $CLASSBODY, $ENUMBODY, $ENUMELEMS, $METHODPARAM}

T = { ENM, MODIFIER, ID, SCOLON, OB, CB, TYPE, METHODBODY, INIT }

R = {

1. $CLASSBODY ➝ENM ID OB $ENUMBODY CB
2. $CLASSBODY ➝ENM ID OB CB
3. $CLASSBODY ➝ MODIFIER TYPE ID INIT SCOLON
4. $CLASSBODY ➝ MODIFIER TYPE ID SCOLON
5. $CLASSBODY ➝ MODIFIER TYPE ID OP $METHODPARAM CP OB METHODBODY CB
6. $CLASSBODY ➝ MODIFIER TYPE ID OP CP OB METHODBODY CB
7. $CLASSBODY ➝ MODIFIER TYPE ID OP $METHODPARAM CP OB CB
8. $CLASSBODY ➝ MODIFIER TYPE ID OP CP OB CB
9. $CLASSBODY ➝ENM ID OB $ENUMBODY CB $CLASSBODY
10. $CLASSBODY ➝ENM ID OB CB $CLASSBODY
11. $CLASSBODY ➝ MODIFIER TYPE ID SCOLON $CLASSBODY
12. $CLASSBODY ➝ MODIFIER TYPE ID INIT SCOLON $CLASSBODY
13. $CLASSBODY ➝ MODIFIER TYPE ID OP $METHODPARAM CP OB METHODBODY CB $CLASSBODY
14. $CLASSBODY ➝ MODIFIER TYPE ID OP CP OB METHODBODY CB $CLASSBODY
15. $CLASSBODY ➝ MODIFIER TYPE ID OP $METHODPARAM CP OB CB $CLASSBODY
16. $CLASSBODY ➝ MODIFIER TYPE ID OP CP OB CB $CLASSBODY
17. $ENUMBODY ➝ID $ENUMELEMS
18. $ENUMBODY ➝ID
19. $ENUMELEMS ➝COMMA $ENUMBODY
20. $METHODPARAM ➝ TYPE ID $NEXTPARAM
21. $METHODPARAM ➝ TYPE ID
22. $NEXTPARAM ➝ COMMA $METHODPARAM

}

Отношения предшествования:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | cb | comma | cp | enm | id | init | methodbody | modifer | ob | op | scolon | type | $ |
| cb |  |  |  | [<.] |  |  |  | [<.] |  |  |  |  |  |
| comma | [.>] |  | [.>] |  | [<.] |  |  |  |  |  |  | [<.] | [.>] |
| cp |  |  |  |  |  |  |  |  | [=.] |  |  |  |  |
| enm |  |  |  |  | [=.] |  |  |  |  |  |  |  |  |
| id | [.>] | [<.] | [.>] |  |  | [=.] |  |  | [=.] | [=.] | [=.] |  | [.>] |
| init |  |  |  |  |  |  |  |  |  |  | [=.] |  |  |
| methodbody | [=.] |  |  |  |  |  |  |  |  |  |  |  |  |
| modifer |  |  |  |  |  |  |  |  |  |  |  | [=.] |  |
| ob | [=.] |  |  |  | [<.] |  | [=.] |  |  |  |  |  |  |
| op |  |  | [=.] |  |  |  |  |  |  |  |  | [<.] |  |
| scolon |  |  |  | [<.] |  |  |  | [<.] |  |  |  |  | [.>] |
| type |  |  |  |  | [=.] |  |  |  |  |  |  |  |  |
| # |  |  |  | [<.] |  |  |  | [<.] |  |  |  |  |  |

Таблица переноса :

|  |  |
| --- | --- |
|  | € |
| cb [enm, id, ob, CLASSBODY, cb] | 0, CLASSBODY |
| scolon [enm, id, ob, CLASSBODY, cb] | 0, CLASSBODY |
| # [enm, id, ob, CLASSBODY, cb] | 0, CLASSBODY |
| cb [enm, id, ob, cb] | 1, CLASSBODY |
| scolon [enm, id, ob, cb] | 1, CLASSBODY |
| # [enm, id, ob, cb] | 1, CLASSBODY |
| cb [modifer, type, id] | 2, CLASSBODY |
| scolon [modifer, type, id] | 2, CLASSBODY |
| # [modifer, type, id] | 2, CLASSBODY |
| cb [modifer, type, id, op, CLASSBODY, cp, ob, methodbody, cb] | 3, CLASSBODY |
| scolon [modifer, type, id, op, CLASSBODY, cp, ob, methodbody, cb] | 3, CLASSBODY |
| # [modifer, type, id, op, CLASSBODY, cp, ob, methodbody, cb] | 3, CLASSBODY |
| cb [modifer, type, id, op, cp, ob, methodbody, cb] | 4, CLASSBODY |
| scolon [modifer, type, id, op, cp, ob, methodbody, cb] | 4, CLASSBODY |
| # [modifer, type, id, op, cp, ob, methodbody, cb] | 4, CLASSBODY |
| cb [modifer, type, id, op, CLASSBODY, cp, ob, cb] | 5, CLASSBODY |
| scolon [modifer, type, id, op, CLASSBODY, cp, ob, cb] | 5, CLASSBODY |
| # [modifer, type, id, op, CLASSBODY, cp, ob, cb] | 5, CLASSBODY |
| cb [modifer, type, id, op, cp, ob, cb] | 6, CLASSBODY |
| scolon [modifer, type, id, op, cp, ob, cb] | 6, CLASSBODY |
| # [modifer, type, id, op, cp, ob, cb] | 6, CLASSBODY |
| cb [enm, id, ob, CLASSBODY, cb, CLASSBODY] | 7, CLASSBODY |
| scolon [enm, id, ob, CLASSBODY, cb, CLASSBODY] | 7, CLASSBODY |
| # [enm, id, ob, CLASSBODY, cb, CLASSBODY] | 7, CLASSBODY |
| cb [enm, id, ob, cb, CLASSBODY] | 8, CLASSBODY |
| scolon [enm, id, ob, cb, CLASSBODY] | 8, CLASSBODY |
| # [enm, id, ob, cb, CLASSBODY] | 8, CLASSBODY |
| cb [modifer, type, id, scolon, CLASSBODY] | 9, CLASSBODY |
| scolon [modifer, type, id, scolon, CLASSBODY] | 9, CLASSBODY |
| # [modifer, type, id, scolon, CLASSBODY] | 9, CLASSBODY |
| cb [modifer, type, id, op, CLASSBODY, cp, ob, methodbody, cb, CLASSBODY] | 10, CLASSBODY |
| scolon [modifer, type, id, op, CLASSBODY, cp, ob, methodbody, cb, CLASSBODY] | 10, CLASSBODY |
| # [modifer, type, id, op, CLASSBODY, cp, ob, methodbody, cb, CLASSBODY] | 10, CLASSBODY |
| cb [modifer, type, id, op, cp, ob, methodbody, cb, CLASSBODY] | 11, CLASSBODY |
| scolon [modifer, type, id, op, cp, ob, methodbody, cb, CLASSBODY] | 11, CLASSBODY |
| # [modifer, type, id, op, cp, ob, methodbody, cb, CLASSBODY] | 11, CLASSBODY |
| cb [modifer, type, id, op, CLASSBODY, cp, ob, cb, CLASSBODY] | 12, CLASSBODY |
| scolon [modifer, type, id, op, CLASSBODY, cp, ob, cb, CLASSBODY] | 12, CLASSBODY |
| # [modifer, type, id, op, CLASSBODY, cp, ob, cb, CLASSBODY] | 12, CLASSBODY |
| cb [modifer, type, id, op, cp, ob, cb, CLASSBODY] | 13, CLASSBODY |
| scolon [modifer, type, id, op, cp, ob, cb, CLASSBODY] | 13, CLASSBODY |
| # [modifer, type, id, op, cp, ob, cb, CLASSBODY] | 13, CLASSBODY |
| comma [id, CLASSBODY] | 14, CLASSBODY |
| ob [id, CLASSBODY] | 14, CLASSBODY |
| comma [id] | 15, CLASSBODY |
| ob [id] | 15, CLASSBODY |
| id [comma, CLASSBODY] | 16, CLASSBODY |
| cb [modifer, type, id, init, scolon] | 17, CLASSBODY |
| scolon [modifer, type, id, init, scolon] | 17, CLASSBODY |
| # [modifer, type, id, init, scolon] | 17, CLASSBODY |
| cb [modifer, type, id, init, scolon, CLASSBODY] | 18, CLASSBODY |
| scolon [modifer, type, id, init, scolon, CLASSBODY] | 18, CLASSBODY |
| # [modifer, type, id, init, scolon, CLASSBODY] | 18, CLASSBODY |
| comma [type, id] | 19, CLASSBODY |
| op [type, id] | 19, CLASSBODY |
| comma [type, id, CLASSBODY] | 21, CLASSBODY |
| op [type, id, CLASSBODY] | 21, CLASSBODY |

Таблица свертки:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | cb | comma | cp | enm | id | init | methodbody | modifer | ob | op | scolon | type | $ |
| [cb, C] |  |  |  | S |  |  |  | S |  |  |  |  | R |
| [comma, C] | R |  | R |  | S |  |  |  |  |  |  | S |  |
| [cp, C] |  |  |  |  |  |  |  |  | S |  |  |  |  |
| enm, C] |  |  |  |  | S |  |  |  |  |  |  |  |  |
| [id, C] | R | S | R |  |  | S |  |  | S | S | S |  | R |
| [init, C] |  |  |  |  |  |  |  |  |  |  | S |  |  |
| [methodbody, C] | S |  |  |  |  |  |  |  |  |  |  |  |  |
| [modifer, C] |  |  |  |  |  |  |  |  |  |  |  |  |  |
| [ob, C] | S |  |  |  | S |  | S |  |  |  |  |  |  |
| [op, C] |  |  | S |  |  |  |  |  |  |  |  |  |  |
| [scolon, C] |  |  |  | S |  |  |  | S |  |  |  |  | R |
| [type, C] |  |  |  |  | S |  |  |  |  |  |  |  |  |
| [#] |  |  |  | S |  |  |  | S |  |  |  |  |  |
| [#, CLASSBODY] |  |  |  | S |  |  |  | S |  |  |  |  | A |

**Subgrammar “Type”**

G3 = <T,N, $TYPE,R>

N = { $TYPE, $PRIMTYPE, $REFTYPE }

T = { TYPE0, TYPE1, TYPE2, TYPE3, TYPE4, TYPE4, TYPE5, TYPE6, TYPE7, TYPE8, ID}

R = {

1. $TYPE ➝$PRIMTYPE
2. $TYPE ➝$REFTYPE
3. $PRIMTYPE ➝ TYPE0
4. $PRIMTYPE ➝ TYPE1
5. $PRIMTYPE ➝ TYPE2
6. $PRIMTYPE ➝ TYPE2
7. $PRIMTYPE ➝ TYPE3
8. $PRIMTYPE ➝ TYPE4
9. $PRIMTYPE ➝ TYPE5
10. $PRIMTYPE ➝ TYPE6
11. $PRIMTYPE ➝ TYPE7
12. $REFTYPE ➝TYPE8
13. $REFTYPE ➝ID

}

Отношения предшествования:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | id | type0 | type1 | type2 | type3 | type4 | type5 | type6 | type7 | type8 | $ |
| id |  |  |  |  |  |  |  |  |  |  | [.>] |
| type0 |  |  |  |  |  |  |  |  |  |  | [.>] |
| type1 |  |  |  |  |  |  |  |  |  |  | [.>] |
| type2 |  |  |  |  |  |  |  |  |  |  | [.>] |
| type3 |  |  |  |  |  |  |  |  |  |  | [.>] |
| type4 |  |  |  |  |  |  |  |  |  |  | [.>] |
| type5 |  |  |  |  |  |  |  |  |  |  | [.>] |
| type6 |  |  |  |  |  |  |  |  |  |  | [.>] |
| type7 |  |  |  |  |  |  |  |  |  |  | [.>] |
| type8 |  |  |  |  |  |  |  |  |  |  | [.>] |
| # | [<.] | [<.] | [<.] | [<.] | [<.] | [<.] | [<.] | [<.] | [<.] | [<.] |  |

Таблица переноса :

|  |  |
| --- | --- |
|  | € |
| # [type0] | 2, TYPE |
| # [type1] | 3, TYPE |
| # [type2] | 4, TYPE |
| # [type3] | 5, TYPE |
| # [type4] | 6, TYPE |
| # [type5] | 7, TYPE |
| # [type6] | 8, TYPE |
| # [type7] | 9, TYPE |
| # [type8] | 10, TYPE |
| # [id] | 11, TYPE |

Таблица свертки:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | id | type0 | type1 | type2 | type3 | type4 | type5 | type6 | type7 | type8 | $ |
| [id, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type0, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type1, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type2, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type3, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type4, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type5, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type6, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type7, C] |  |  |  |  |  |  |  |  |  |  | R |
| [type8, C] |  |  |  |  |  |  |  |  |  |  | R |
| [#] | S | S | S | S | S | S | S | S | S | S |  |
| [#, TYPE] | S | S | S | S | S | S | S | S | S | S | A |

**Subgrammar “Initialize block”**

G4 = <T,N, $INIT,R>

N = { $INIT, $VALUE, $MATREXP, $MATREXPBODY, $MATREXPELEM }

T = { ASSIGN, INT, BOOL, REAL, NEW, TYPE8, OBT, INT, CBT, OB, CB, ID, COMMA, OP, CP, EXRESSION }

R = {

(0) INIT -> ASSIGN EXRESSION

(1) INIT -> ASSIGN MATREXP

(2) VALUE -> INT

(3) VALUE -> BOOL

(4) VALUE -> REAL

(5) MATREXP -> NEW TYPE8 OBT INT CBT OBT INT CBT

(6) MATREXP -> NEW TYPE8 OBT INT CBT

(7) MATREXP -> OB MATREXPBODY CB

(8) MATREXPBODY -> OB MATREXPELEM CB comma OB MATREXPELEM CB

(9) MATREXPELEM -> VALUE

(10) MATREXPELEM -> VALUE COMMA MATREXPELEM

(11) VALUE -> ID

}

Отношения предшествования:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | assign | bool | cb | cbt | comma | expression | id | int | new | ob | obt | real | type8 | $ |
| assign |  |  |  |  |  | [=.] |  |  | [<.] | [<.] |  |  |  | [.>] |
| bool |  |  | [.>] |  | [.>] |  |  |  |  |  |  |  |  |  |
| cb |  |  | [.>] |  | [=.] |  |  |  |  |  |  |  |  | [.>] |
| cbt |  |  |  |  |  |  |  |  |  |  | [=.] |  |  | [.>] |
| comma |  | [<.] | [.>] |  | [<.] |  | [<.] | [<.] |  | [=.] |  | [<.] |  |  |
| expression |  |  |  |  |  |  |  |  |  |  |  |  |  | [.>] |
| id |  |  | [.>] |  | [.>] |  |  |  |  |  |  |  |  |  |
| int |  |  | [.>] | [=.] | [.>] |  |  |  |  |  |  |  |  |  |
| new |  |  |  |  |  |  |  |  |  |  |  |  | [=.] |  |
| ob |  | [<.] | [=.] |  | [<.] |  | [<.] | [<.] |  | [<.] |  | [<.] |  |  |
| obt |  |  |  |  |  |  |  | [=.] |  |  |  |  |  |  |
| real |  |  | [.>] |  | [.>] |  |  |  |  |  |  |  |  |  |
| type8 |  |  |  |  |  |  |  |  |  |  | [=.] |  |  |  |
| # | [<.] |  |  |  |  |  |  |  |  |  |  |  |  |  |

Таблица переноса :

|  |  |
| --- | --- |
|  | € |
| # [assign, expression] | 0, INIT |
| # [assign, INIT] | 1, INIT |
| comma [int] | 2, INIT |
| ob [int] | 2, INIT |
| comma [bool] | 3, INIT |
| ob [bool] | 3, INIT |
| comma [real] | 4, INIT |
| ob [real] | 4, INIT |
| assign [new, type8, obt, int, cbt, obt, int, cbt] | 5, INIT |
| assign [new, type8, obt, int, cbt] | 6, INIT |
| assign [ob, INIT, cb] | 7, INIT |
| ob [ob, INIT, cb] | 7, INIT |
| assign [ob, INIT, cb, comma, ob, INIT, cb] | 8, INIT |
| ob [ob, INIT, cb, comma, ob, INIT, cb] | 8, INIT |
| comma [INIT, comma, INIT] | 10, INIT |
| ob [INIT, comma, INIT] | 10, INIT |
| comma [id] | 11, INIT |
| ob [id] | 11, INIT |

Таблица свертки:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | assign | bool | cb | cbt | comma | expression | id | int | new | ob | obt | real | type8 | $ |
| [assign, C] |  |  |  |  |  | S |  |  | S | S |  |  |  | R |
| [bool, C] |  |  | R |  | R |  |  |  |  |  |  |  |  |  |
| [cb, C] |  |  | R |  | S |  |  |  |  |  |  |  |  | R |
| [cbt, C] |  |  |  |  |  |  |  |  |  |  | S |  |  | R |
| [comma, C] |  | S | R |  | S |  | S | S |  | S |  | S |  |  |
| [expression, C] |  |  |  |  |  |  |  |  |  |  |  |  |  | R |
| [id, C] |  |  | R |  | R |  |  |  |  |  |  |  |  |  |
| [int, C] |  |  | R | S | R |  |  |  |  |  |  |  |  |  |
| [new, C] |  |  |  |  |  |  |  |  |  |  |  |  | S |  |
| [ob, C] |  | S | S |  | S |  | S | S |  | S |  | S |  |  |
| [obt, C] |  |  |  |  |  |  |  | S |  |  |  |  |  |  |
| [real, C] |  |  | R |  | R |  |  |  |  |  |  |  |  |  |
| [type8, C] |  |  |  |  |  |  |  |  |  |  | S |  |  |  |
| [#] | S |  |  |  |  |  |  |  |  |  |  |  |  |  |
| [#, INIT] | S |  |  |  |  |  |  |  |  |  |  |  |  | A |

**Subgrammar “Expressions”**

G5 = <T,N, $EXRESSION,R>

N = { $EXRESSION, $TERNARYEXP, $LOGICEXP, $MATREXPBODY, $MATREXPELEM }

T = { REAL, BOOL, INT, ID, OP, CP, QTO, CTO, MULT, ADDIT, LOGIC, COMPARE, HEIGHT, WEIGHT, transpose }

R = {

(15) EXPRESSION -> TERNARYEXP

(16) TERNARYEXP -> LOGICEXP QTO LOGICEXP CTO LOGICEXP

(17) TERNARYEXP -> LOGICEXP

(18) LOGICEXP -> EQUEXP

(19) EQUEXP -> ADDEXPRES

(20) ADDEXPRES -> MULTEXP

(21) MULTEXP -> PRIMARY

(0) EQUEXP -> ADDEXPRES COMPARE ADDEXPRES

(1) ADDEXPRES -> MULTEXP ADDIT MULTEXP

(2) MULTEXP -> PRIMARY MULT PRIMARY

(3) PRIMARY -> OP EXPRESSION CP

(4) PRIMARY -> NUMBER

(5) PRIMARY -> VAR

(6) NUMBER -> INT

(7) NUMBER -> REAL

(8) NUMBER -> BOOL

(9) VAR -> ID

(10) LOGICEXP -> EQUEXP LOGIC EQUEXP

(11) PRIMARY -> MATROPER

(12) MATROPER -> HEIGHT OP VAR CP

(13) MATROPER -> WEIGHT OP VAR CP

(14) MATROPER -> transpose OP VAR CP

}

Отношения предшествования:

И другие таблицы…

**Subgrammar “Method Body”**

G6 = <T,N, $METHODBODY,R>

N = { $METHODBODY, $OPERATOR, $VARDECBODY, $WHILE, $WHILE }

T = { RETURN, SCOLON, MODIFIER, ID, EXRESSION, TYPE, OB , WHILE, CB, OP, CP }

R = {

$METHODBODY ➝ $OPERATOR

$METHODBODY ➝ $OPERATOR RETURN SCOLON

$METHODBODY ➝ $OPERATOR RETURN $EXRESSION SCOLON

$OPERATOR ➝ MODIFIER $VARDECBODY SCOLON

$OPERATOR ➝ $VARDECBODY SCOLON

$OPERATOR ➝ $EXRESSION SCOLON

$OPERATOR ➝ $WHILE SCOLON

$VARDECBODY ➝ $TYPE ID $INIT

$VARDECBODY ➝ $TYPE ID

$OPERATOR ➝ MODIFIER $VARDECBODY SCOLON $OPERATOR

$OPERATOR ➝ $VARDECBODY SCOLON $OPERATOR

$OPERATOR ➝ $EXRESSION SCOLON $OPERATOR

$OPERATOR ➝ $WHILE SCOLON $OPERATOR

$WHILE ➝ DO OB $METHODBODY CB WHILE OP EXPRESSION CP

$WHILE ➝ DO OB CB WHILE OP EXPRESSION CP

}

Отношения предшествования:

И другие таблицы…