|  |  |
| --- | --- |
| **Код** | **Лексема** |
| 1 | program |
| 2 | int |
| 3 | real |
| 4 | label |
| 5 | begin |
| 6 | cout |
| 7 | cin |
| 8 | for |
| 9 | to |
| 10 | by |
| 11 | while |
| 12 | end |
| 13 | end. |
| 14 | and |
| 15 | or |
| 16 | not |
| 17 | if |
| 18 | then |
| 19 | goto |
| 20 | , |
| 21 | + |
| 22 | - |
| 23 | \* |
| 24 | / |
| 25 | ( |
| 26 | ) |
| 27 | : |
| 28 | < |
| 29 | > |
| 30 | = |
| 31 | ¶ |
| 32 | << |
| 33 | >> |
| 34 | <= |
| 35 | >= |
| 36 | == |
| 37 | != |
| 38 | idn |
| 39 | con |

Таблиця лексем:

Завдання (84221)

Приклад програми:

1. program example
2. int i,j
3. real a,b
4. label m1
5. begin
6. a=1
7. j=1
8. for i=0 to 8 by i+1 while(j<=5) j=j+1
9. a=a\*3
10. end
11. cin>>b
12. if b<a then goto m1
13. m1: cout<<a
14. end.

Вихідна таблиця лексем:

| **№** | **№ рядка** | **Лексема** | **Код лексеми** | **Код idn/con** |
| --- | --- | --- | --- | --- |
| 1 | 1 | program | 1 |  |
| 2 | 1 | example | 38 | 1 |
| 3 | 1 | ¶ | 31 |  |
| 4 | 2 | int | 2 |  |
| 5 | 2 | i | 38 | 2 |
| 6 | 2 | , | 20 |  |
| 7 | 2 | j | 38 | 3 |
| 8 | 2 | ¶ | 31 |  |
| 9 | 3 | real | 3 |  |
| 10 | 3 | a | 38 | 4 |
| 11 | 3 | , | 20 |  |
| 12 | 3 | b | 38 | 5 |
| 13 | 3 | ¶ | 31 |  |
| 14 | 4 | label | 4 |  |
| 15 | 4 | m1 | 38 | 6 |
| 16 | 4 | ¶ | 31 |  |
| 17 | 5 | begin | 5 |  |
| 18 | 5 | ¶ | 31 |  |
| 19 | 6 | a | 38 | 4 |
| 20 | 6 | = | 30 |  |
| 21 | 6 | 1 | 39 | 1 |
| 22 | 6 | ¶ | 31 |  |
| 23 | 7 | j | 38 | 3 |
| 24 | 7 | = | 30 |  |
| 25 | 7 | 1 | 39 | 1 |
| 26 | 7 | ¶ | 31 |  |
| 27 | 8 | for | 8 |  |
| 28 | 8 | i | 38 | 2 |
| 29 | 8 | = | 30 |  |
| 30 | 8 | 0 | 39 | 2 |
| 31 | 8 | to | 9 |  |
| 32 | 8 | 8 | 39 | 3 |
| 33 | 8 | by | 10 |  |
| 34 | 8 | i | 38 | 2 |
| 35 | 8 | + | 21 |  |
| 36 | 8 | 1 | 39 | 1 |
| 37 | 8 | while | 11 |  |
| 38 | 8 | ( | 25 |  |
| 39 | 8 | j | 38 | 3 |
| 40 | 8 | <= | 34 |  |
| 41 | 8 | 5 | 39 | 4 |
| 42 | 8 | ) | 26 |  |
| 43 | 8 | j | 38 | 3 |
| 44 | 8 | = | 30 |  |
| 45 | 8 | j | 38 | 3 |
| 46 | 8 | + | 21 |  |
| 47 | 8 | 1 | 39 | 1 |
| 48 | 8 | ¶ | 31 |  |
| 49 | 9 | a | 38 | 4 |
| 50 | 9 | = | 30 |  |
| 51 | 9 | a | 38 | 4 |
| 52 | 9 | \* | 23 |  |
| 53 | 9 | 3 | 39 | 5 |
| 54 | 9 | ¶ | 31 |  |
| 55 | 10 | end | 12 |  |
| 56 | 10 | ¶ | 31 |  |
| 57 | 11 | cin | 7 |  |
| 58 | 11 | >> | 33 |  |
| 59 | 11 | b | 38 | 5 |
| 60 | 11 | ¶ | 31 |  |
| 61 | 12 | if | 17 |  |
| 62 | 12 | b | 38 | 5 |
| 63 | 12 | < | 28 |  |
| 64 | 12 | a | 38 | 4 |
| 65 | 12 | then | 18 |  |
| 66 | 12 | goto | 19 |  |
| 67 | 12 | m1 | 38 | 6 |
| 68 | 12 | ¶ | 31 |  |
| 69 | 13 | m1 | 38 | 6 |
| 70 | 13 | : | 27 |  |
| 71 | 13 | cout | 6 |  |
| 72 | 13 | << | 32 |  |
| 73 | 13 | a | 38 | 4 |
| 74 | 13 | ¶ | 31 |  |
| 75 | 14 | end. | 13 |  |

Таблиця ідентифікаторів:

|  |  |  |
| --- | --- | --- |
| **Код** | **ІД** | **Тип** |
| 1 | example | program |
| 2 | i | int |
| 3 | j | int |
| 4 | a | real |
| 5 | b | real |
| 6 | m1 | label |

Таблиця констант:

|  |  |
| --- | --- |
| **Код** | **Лексема** |
| 1 | 1 |
| 2 | 0 |
| 3 | 8 |
| 4 | 5 |
| 5 | 3 |

**Граматика**

1. <програма> ::= program <назва програми>¶ <список огол.> begin¶ <список операторів> end.
2. <назва програми> ::= <ідент.>
3. <список огол.> ::= <тип> <список ідент.> ¶ **|** <список огол.> <тип> <список ідент.> ¶
4. <тип> ::= int **|** real **|** label
5. <список ідент.> ::= <ідент.> **|** <список ідент.> , <ідент.>
6. <список операторів> ::= <оператор>¶ **|** <список операторів> <оператор>¶
7. <оператор> ::= <основні оператори> **|** <ідент.>: <основні оператори>
8. <основні оператори> ::= <вивід> **|** <ввід> **|** <присвоєння> **|** <цикл> **|** <умовний перехід> **|** <оператор безумовного переходу>
9. <оператор безумовного переходу> ::= goto <ідент.>
10. <оператор присвоєння> ::= <ідент.> = <E>
11. <вивід> ::= <вивід> <оператор вивед.> **|** cout <оператор вивед.>
12. <оператор вивед.> ::= << <ідент.> **|** << <конст.>
13. <ввід> ::= <ввід> <оператор введ.> **|** cin <оператор введ.>
14. <оператор введ.> ::= >> <ідент.>
15. <умовний перехід> ::= if <відношення> then <оператор>
16. <цикл> ::= for <ідент.> = <E> to <E> by <E> while (<лог. вираз>) <список операторів> end
17. <лог. вираз> ::= <лог. терм> **|** <лог. вираз> or <лог. терм>
18. <лог. терм> ::= <лог. множник> **|** <лог. терм> and <лог. множник>
19. <лог. множник> ::= <відношення> **|** (<лог. вираз>) **|** not <лог. множник>
20. <відношення> ::= <E> <знак відношення> <E>
21. <знак відношення> ::= < **|** > **|** == **|** <= **|** >= **|** !=
22. <E> ::= <T> **|** <E> + <T> **|** <E> - <T>
23. <T> ::= <o> **|** <T> \* <o> **|** <T> / <o>
24. <o> ::= <конст.> **|** <ідент.> **|** (<E>)
25. <ідент.> ::= <буква> **|** <ідент.> <буква> **|** <ідент.> <цифра>
26. <буква> ::= a **|** … **|** z **|**A **|** … **|**Z **|** \_
27. <цифра> ::= 0 **|** … **|** 9
28. <конст.> ::= <знак> <ц.б.з.> **|** <знак> . <ц.б.з.> **|** <знак> <ц.б.з.>. **|** <знак> <ц.б.з.> . <ц.б.з.>
29. <знак> ::= + **|** - **|** /\
30. <ц.б.з.> ::= <цифра> **|** <ц.б.з.> <цифра>

МПА у формі таблиці переходів

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| α | мітка | β | стек | п/п по [!=] | п/п по [=] |
| 1 | program | 2 |  | error |  |
| 2 | IDN | 3 |  | error |  |
| 3 | ¶ | 4 |  | error |  |
| 4 | int|real|label | 5 |  | error |  |
| 5 | IDN | 6 |  | error |  |
| 6 | ,  ¶ | 5  7 |  | error |  |
| 7 | begin  int|real|label | 8  5 |  | error |  |
| 8 | ¶ | п/а <опер.> 11 | ↓9 | error |  |
| 9 | ¶ | 10 |  | error |  |
| 10 | end. |  |  | п/а <опер.> 11 ↓9 | exit |
| 11 | IDN  cout  cin  for  if  goto | 12  16  19  22  п/а <E> 35  34 | ↓31 | error |  |
| 12 | =  : | п/а <E> 35  13 | ↓15 | error |  |
| 13 | IDN  cout  cin  for  if  goto | 14  16  19  22  п/а <E> 35  34 | ↓31 | error |  |
| 14 | = | п/а <E> 35 | ↓15 | error |  |
| 15 | @ |  |  | exit |  |
| 16 | << | 17 |  | error |  |
| 17 | IDN|CON | 18 |  | error |  |
| 18 | <<  @ | 17 |  | exit |  |
| 19 | >> | 20 |  | error |  |
| 20 | IDN | 21 |  | error |  |
| 21 | >>  @ | 20 |  | exit |  |
| 22 | IDN | 23 |  | error |  |
| 23 | = | п/а <E> 35 | ↓24 | error |  |
| 24 | to | п/а <E> 35 | ↓25 | error |  |
| 25 | by | п/а <E> 35 | ↓26 | error |  |
| 26 | while | 27 |  | error |  |
| 27 | ( | п/а <лог. вираз> 38 | ↓28 | error |  |
| 28 | ) | п/а <опер.> 11 | ↓29 | error |  |
| 29 | ¶ | 30 |  | error |  |
| 30 | end |  |  | п/а <опер.> 11 ↓29 | exit |
| 31 | < <= == != >= > | п/а <E> 35 | ↓32 | error |  |
| 32 | then | п/а <опер.> 11 | ↓33 | error |  |
| 33 | @ |  |  |  | exit |
| 34 | IDN |  |  | error | exit |
| 35 | IDN|CON  ( | 37  п/а <E> 35 | ↓36 | error |  |
| 36 | ) | 37 |  | error |  |
| 37 | + - \* / | 35 |  | exit |  |
| 38 | not  ( | 38  п/а <лог. вираз> 38 | ↓40 | п/а <E> 35 ↓39 |  |
| 39 | < <= == != >= > | п/а <E> 35 | ↓41 | error |  |
| 40 | ) | 41 |  | error |  |
| 41 | and|or | 38 |  | exit |  |

**Граматика(рекурсивний спуск)**

1. <програма> ::= program IDN ¶ <список огол.> begin¶ <список операторів> end.
2. <список огол.> ::= <тип> <список ідент.> ¶ {<тип> <список ідент.> ¶ }
3. <тип> ::= int **|** real **|** label
4. <список ідент.> ::= IDN{ , IDN }
5. <список операторів> ::= <оператор>¶ { <оператор>¶ }
6. <оператор> ::= IDN: <осн. оператори> **|** <осн. оператори>
7. <осн. оператори> ::= cout << (IDN|CON){<< (IDN|CON)} **|** cin >> IDN {>> IDN} **|** IDN = <E> **|** for IDN = <E> to <E> by <E> while (<лог. вираз>) <список операторів> end **|** if <відношення> then <оператор>**|** goto IDN
8. <лог. вираз> ::= <лог. терм> {or <лог. терм> }
9. <лог. терм> ::= <лог. множник> {and <лог. множник> }
10. <лог. множник> ::= <відношення> **|** (<лог. вираз>) **|** not <лог. множник>
11. <відношення> ::= <E> <знак відношення> <E>
12. <знак відношення> ::= < **|** > **|** == **|** <= **|** >= **|** !=
13. <E> ::= <T> { + <T> **|** - <T> }
14. <T> ::= <o> { \* <o> **|** / <o> }
15. <o> ::= CON **|** IDN **|** (<E>)

**Граматика простого передування**

1. <програма>::=program IDN <список\_огол1> begin <список\_опер1> end.
2. <список\_огол1>::=<список\_огол>
3. <список\_огол>::=<тип> <список\_ід1> ¶|<список\_огол> <тип> <список\_ід1> ¶
4. <тип>::=int|real|label
5. <список\_ід1>::=<список\_ід>
6. <список\_ід>::=, IDN|<список\_ід> , IDN
7. <список\_опер1>::=<список\_опер>
8. <список\_опер>::=<опер1> ¶|<список\_опер> <опер1> ¶
9. <опер1>::=<опер>
10. <опер>::=<осн\_опер>|IDN : <осн\_опер>
11. <осн\_опер>::=<вивід>|<ввід>|<присвоєння>|<цикл>|<ум\_перехід>|<опер\_безум\_переходу>
12. <опер\_безум\_переходу>::=goto IDN
13. <присвоєння>::=IDN = <E1>
14. <вивід>::=<вивід> <опер\_вивед>|cout <опер\_вивед>
15. <опер\_вивед>::=<< IDN|<< CON
16. <ввід>::=<ввід> <опер\_введ>|cin <опер\_введ>
17. <опер\_введ>::=>> IDN
18. <ум\_перехід>::=if <відн> then <опер>
19. <цикл>::=for IDN = <E1> to <E1> by <E1> while { <лог\_вираз1> } <список\_опер1> end
20. <лог\_вираз1>::=<лог\_вираз>
21. <лог\_вираз>::=<лог\_терм1>|<лог\_вираз> or <лог\_терм1>
22. <лог\_терм1>::=<лог\_терм>
23. <лог\_терм>::=<лог\_мн1>|<лог\_терм> and <лог\_мн1>
24. <лог\_мн1>::=<лог\_мн>
25. <лог\_мн>::=<відн>|[ <лог\_вираз1> ]|not <лог\_мн>
26. <відн>::=<E1> <зн\_відн> <E1>
27. <зн\_відн>::=<|>|==|<=|>=|!=
28. <E1>::=<E>
29. <E>::=<T1>|<E> + <T1>|<E> - <T1>
30. <T1>::=<T>
31. <T>::=<o>|<T> \* <o>|<T> / <o>
32. <o>::=IDN|CON|( <E1> )