***Программа:*** *главная программа языка С++.*

*Допускается описание массивов как типов (typedef).*

*Разрешается описание глобальных данных.*

***Типы данных:*** *int и long int.*

***Операции:*** *арифметические, сравнения.*

***Операторы:*** *присваивания и while, пустой и составной оператор.*

***Операнды:*** *простые переменные, элементы массивов и константы.*

***Константы:*** *целые в 10 c/c и 16 c/c .*

<Программа> -> < Программа ><описание> | < Программа ><оператор> | ε

<описание > -><описание данных> | <массив>

<массив> -> typedef <тип > < идентификатор > <размер> | typedef <тип > < идентификатор > <размер> = {<список элементов массива>}

< список элементов массива> -> < список элементов массива>, <константа> | <константа >

<размер> -> [<константа целая в 10с/с >]

< описание данных > -><тип ><список переменных>;

<тип> -> int | long int

<список переменных> -><список переменных>, <переменная> | <переменная>

< переменная> -><идентификатор > | < идентификатор > = <выражение>

< составной оператор > ->{<описание и операторы>}

<описание и операторы> -> <описание и операторы> < описание данных >| <описание и операторы> <оператор>| ε

//Здесь описание данных схоже с описанием глобальных данных, поэтому используем тот же нетерминал

<оператор> -><присваивания>|<составной оператор>|<while>| ;

//;-пустой оператор

Составной оператор – грубо говоря, список операторов в фигурных скобках. Поэтому я использовал его в качестве тела функции и оператора while

<присваивания> -><идентификатор>=<выражение>;

<while> ->while (<выражение>) <оператор>;

/\* Уровни приоритета операций:

1. Все операции сравнения (< выражение >)
2. Бинарные “+” и “–”(<слагаемое>)
3. Операции “\*” и “/”(<множитель>)

\*/

<параметры>-><выражение >><слагаемое>|

<выражение >>= <слагаемое>|

<выражение ><<слагаемое>|

<выражение ><= <слагаемое>|

<выражение > == <слагаемое>|

<выражение >!= <слагаемое>

<слагаемое> -><слагаемое>+<множитель> |

<слагаемое>-<множитель> |

<множитель>

<множитель> -><множитель>\*<эл. выражение> |

<множитель>/<эл. выражение> |

<эл. выражение>

<эл. выражение>-><идентификатор>|<константа>| <элемент массива>|(<выражение>)

//В задании указано, операнды должны являться простыми переменными и константами

<элемент массива> -> <идентификатор><размер>

<идентификатор>-><буква><окончание>

<окончание>-><окончание><буква>|<окончание><цифра>| ε

<константа> -> <константа целая в 10с/с>|<константа целая в 16c/c>

<константа целая в 10с/с> - <константа целая в 10с/с><цифра>|<цифра>

<константа целая в 16c/c> - <константа целая в 16c/c><16цифра>|<16цифра>

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

<16цифра> -> 0|1|...|9|A|B|…|F|a|b|c|...|f

<буква>->a|b|c|…|Z

|  |  |  |
| --- | --- | --- |
| Лексические единицы языка | Тип лексемы | Символ-ограничитель |
| **Ключевые слова** |  |  |
| While  void | TWhile = 1  TVoid = 2 | Не буква, не цифра  Не буква, не цифра |
| int | TInt = 3 | Не буква, не цифра |
| Long | Tlong = 4 | Не буква, не цифра |
| **Идентификаторы** | TIdent = 20 | не буква, не цифра |
| Константа целая в 10 с/с | TConstInt10 = 30 | Не цифра, не буква a-f, A-F |
| Константа целая в 16 с/с  **Специальные знаки**  **;**  **,**  **[**  **]**  **{**  **}**  **(**  **)**  **Знаки операций**  \*  /  +  -  <  <=  >  >=  ==  !=  = | TConstInt16 = 31  TTZpt = 40  TZpt = 41  TLKSk = 42  TPKSk = 43  TLFSk = 44  TPFSk = 45  TLSk = 46  TPSk = 47  TMult = 50  TDiv = 51  TPlus = 52  TMinus = 53  TSmall = 54 TSmallEqual = 55 TLarg = 56 TLargEqual = 57 TLogEqual = 58 TNEqual = 59  TEqual = 60 | Не цифра  Любой символ  Любой символ  Любой символ  Любой символ  Любой символ  Любой символ  Любой символ  Любой символ  Любой символ  Любой символ  Любой символ  Любой символ  Не знак =  Любой символ  Не знак =  Любой символ  Любой символ  Любой символ  Не знак = |
| **Конец исходного модуля**  **Ошибочный символ** | TEnd = 100  TErr = 200 |  |

