S <объявления><программа>

S <программа>

<цикл1> : L for V = <to expr> <программа>

<цикл2> : L for V = <to expr> <adding expr> <программа>

<программа case> : <программа>

<программа> <оператор><программа>

<программа> <оператор>

<объявления> → <объявлениe> ;

<объявлениe> → declare V as <тип> <объяв> // , V as <тип><объяв>

<объявлениe> → declare V as <тип>

<объяв> → , V as <тип> <объяв>

<объяв> → , V as <тип>

<тип> → int

<тип> → long

<объявления> → <объявлениe> ;

<объявлениe> → declare V <тип объяв> , V <объяв> // , V as <тип><объяв>

<объявлениe> → declare V <тип объяв>

<тип объяв> → as <тип>

<объяв> → <тип объяв> , V <объяв>

<объяв> → <тип объяв>

<тип> → int

<тип> → long

<объявление> → declare V as <тип> <объяв>

<объяв> → <запятая> V as <тип><объяв> // , V as <тип><объяв>

<объяв> → ;

<тип> → int

<тип> → long

<оператор> ;

, = V V < , или V > , V > to — Надо так

<оператор> <список1> ;

<список1> <set> <переменные> <кон сет>

<список1> <set> <кон сет>

<set> set V

<переменные> <переменные2>

<переменные2> <переменные2><переменные3>

<переменные2> <переменные3>

<переменные3> , V

<кон сет> to <E>

<оператор> <список1> ;

<список1> set V <переменные> <кон сет>

<список1> <set> <кон сет>

<set> set V

<переменные> <переменные2>

<переменные2> , V <переменные>

<переменные2> , V

<кон сет> to <E>

<оператор> <список1> ;

<список1> set V , <переменные>

<список1> <set> to <E>

<set> set V

<переменные> V , <переменные> // , V <переменные>

<переменные> V to <E> // , V

<оператор> set <список1> to <E> ; // <выражение для грам> ;

<выражение присв> <E>

<оператор> set <список1> <выражение присв> ;

<выражение присв> to <E>

<список1> V <переменные>

<список1> V

<переменные> <запятая> V <переменные> // , V <переменные>

<переменные> <запятая> V // , V

<оператор> <цикл1> <next>

<next> next L

<тело цикла> <программа>

<цикл1> : L for V = <to expr> <программа>

<оператор> <цикл2> <next>

<цикл2> : L for V = <to expr> <adding expr> <программа>

<to expr> <E> to <E>

<adding expr> adding <E>

<оператор> if <тест> goto <метка кон> else goto <метка кон> ;

<оператор> if <тест> goto <метка кон> ;

// <если> if <тест> goto L // goto <метка>

// <иначе> else goto L // goto <метка>

<оператор> <ввод перем> cin ;

<ввод перем> <список2>

<список2> V <- <список2>

<список2> V <-

<оператор> <вывод выраж> cout ;

<вывод выраж> <список3>

<список3> <E> -> <список3>

<список3> <E> ->

<оператор> : L

<оператор> goto <метка кон> ; // goto L ;

<константа> > : —— Надо

<оператор> switch ( <E> ) <тело switch> end switch ;

<тело switch> <кейcы>

<кейсы> <кейс><кейcы>

<кейсы> <дефолт>

<кейс> <case Beg> <программа case> break ;

<дефолт> default <программа case> break ;

<case Beg> case <константа>

<оператор> fail ;

<оператор> rem

<тест> <E> отношение <E>

<E> → <T>

<E> → <T> <E′>

<E′> → + <T> <E′>

<E′> → - <T> <E′>

<E′> → + <T>

<E′> → - <T>

<T> → <F>

<T> → <F> <T′>

<T′> → \* <F> <T′>

<T′> → / <F> <T′>

<T′> → % <F> <T′>

<T′> → \* <F>

<T′> → / <F>

<T′> → % <F>

<F> → ( <E> )

<F> → <константа>

<F> → <var>

<F> → digit ( <центр функ> ) V < , C > ) ( < V

<центр функ> → <var> , <константа>

<var> → V

<константа> → C

<константа> → LC

<метка кон> L

<центр функ> → V <откуда>

<откуда> → , <константа>

//<To> → to

//<запятая2> → <запятая>

//<L> → LV // Проверка что переменная является длинного типа

//<L> → LC

//<запятая> → ,