Remove all left recursions		
<prog>	->	program <identifier_start> ; var <dec-list> begin <stat-list> end.
<identifier_start>	->	<letter> <identifier_body>
<identifier_body>	->	<letter> <identifier_body>
<identifier_body>	->	<digit> <identifier_body>
<identifier_body>	->	lamda
<dec-list>	->	<dec> : <type> ;
<dec>	->	<identifier_start> , <dec>
<dec>	->	<identifier_start>
<type>	->	integer
<stat-list>	->	<stat>
<stat-list>	->	<stat> <stat-list>
<stat>	->	<write>
<stat>	->	<assign>
<write>	->	display ("value" , <identifier_start>) ;
<write>	->	display (<identifier_start>) ;
<assign>	->	<identifier_start> = <expr_term_factor_enter> ;
<expr_term_factor_enter>	->	<term_enter> <expr>
<expr>	->	+ <term_enter> <expr>
<expr>	->	- <term_enter> <expr>
<expr>	->	lamda
<term_enter>	->	<factor> <term>
<term>	->	* <factor> <term>
<term>	->	/<factor> <term>
<term>	->	lamda
<factor>	->	(<expr_term_factor_enter>)
<factor>	->	<identifier_start>
<factor>	->	<number>
<number_start>	->	<sign> <digit> <number_body>
<number_body>	->	<digit> <number_body>
<number_body>	->	lamda
<sign>	->	+
<sign>	->	-
<sign>	->	lamda
<digit>	->	0
<digit>	->	1
<digit>	->	2
<digit>	->	3
<digit>	->	4
<digit>	->	5
<digit>	->	6
<digit>	->	7
<digit>	->	8
<digit>	->	9
<letter>	->	p
<letter>	->	q

<letter>	->	r
<letter>	->	s

Remove all Indeterminates		
<prog>	->	program <identifier_start> ; var <dec-list> begin <stat-list_enter> end.
<identifier_start>	->	<letter> <identifier_body>
<identifier_body>	->	<letter> <identifier_body>
<identifier_body>	->	<digit> <identifier_body>
<identifier_body>	->	lamda
<dec-list>	->	<dec_enter> : <type> ;
<dec_enter>	->	<identifier_start> <dec>
<dec>	->	, <dec_enter>
<dec>	->	lamda
<type>	->	integer
<stat-list_enter>	->	<stat> <stat-list>
<stat-list>	->	lamda
<stat-list>	->	<stat> <stat-list_enter>
<stat>	->	<write_enter>
<stat>	->	<assign>
<write_enter>	->	display (<write>
<write>	->	"value" , <identifier_start>) ;
<write>	->	<identifier_start>) ;
<assign>	->	<identifier_start> = <expr_term_factor_enter> ;
<expr_term_factor_enter>	->	<term_enter> <expr>
<expr>	->	+ <term_enter> <expr>
<expr>	->	- <term_enter> <expr>
<expr>	->	lamda
<term_enter>	->	<factor> <term>
<term>	->	* <factor> <term>
<term>	->	/<factor> <term>
<term>	->	lamda
<factor>	->	(<expr_term_factor_enter>)
<factor>	->	<identifier_start>
<factor>	->	<number_start>
<number_start>	->	<sign> <digit> <number_body>
<number_body>	->	<digit> <number_body>
<number_body>	->	lamda
<sign>	->	+
<sign>	->	-
<sign>	->	lamda
<digit>	->	0
<digit>	->	1
<digit>	->	2
<digit>	->	3
<digit>	->	4
<digit>	->	5
<digit>	->	6
<digit>	->	7
<digit>	->	8

<digit>	->	9
<letter>	->	p
<letter>	->	q
<letter>	->	r
<letter>	->	s