• Matching identifiers

Letter = [a-z]|[A-Z]

Digit = [0-9]

Identifier = letter (letter|_|digit)* , identifier | letter (letter|_|digit)*

• Matching reserved words

Yesif-Otherwise-------- > Condition

RepeatWhen/Reiterate-------- > Loop

regular expressions

Yesif-Otherwise -------- > Condition

RepeatWhen/Reiterate -------- > Loop

(+,-,*,/)-------- >Arithmetic Operation

(&&, ||, ~)-------- >Logic operators

(==, , !=, <=, >=) -------- >relational operators

(=)-------- >Assignment operator

(->)-------- >Access Operator

(,),[,]{,} -------- >Braces

“,’ -------- >Quotation Mark

/@-------- >Comment

@/-------- >Comment

/^ -------- >Comment

$-------- >Token Delimiter

. -------- >Line Delimiter

Omw -------- >Integer

SIMww -------- >SInteger

Chji -------- >Character

Seriestl -------- >String

IMwf -------- >Float

SIMwf -------- >SFloat

NOReturn-------- >Void

GetBack-------- >Return

OutLoop-------- >Break

Loli -------- >Struct

Include -------- >Inclusion

Start -------- >Start

Last -------- >End

All ----→∑{+,-,*,/, &&, ||, ~,==, , !=, <=, >=,=,->, (,) , [,] { , } , -> , “,’ , /@ ,@/ ,/^ , $, Include ,Start ,Last ,Omw , SIMww , Chji , Seriestl , IMwf , SIMwf , Yesif-Otherwise, RepeatWhen, Reiterate}

Identifier = letter (letter|_|digit)* , identifier | letter (letter|_|digit)*

Type ----→ Omw , SIMww , Chji , Seriestl , IMwf , SIMwf

Integers: [+]?[0-9]+

Floats: [+]?(([0-9]+ (.[0-9]⇤)?|.[0-9]+)([eE][+]?[0-9]+)?)

String constants: “([a-zA-Z0-9]|\[a-zA-Z])⇤”

Expr = ()

Statement = identifier

L(a) = identifier

regular expression of Start

R = Start

regular expression of define var

R = (ε |Type) (identifier | ,)+ (ε | = (Constant| identifier)).

regular expression of Condition

R = Yesif (identifier relational operators (identifier | Constant))

{

(Statement (. | Last))*

}

Otherwise

{

(Statement (. | Last))*

}

regular expression of Function

R = (Type | void) identifier (ε | (Type identifier)*)

{

(Statement (. | Last))*

(ε | GetBack (. | Last))

}

regular expression of Loop

Reiterate

{

(Statement (. | Last))*

(ε | OutLoop (. | Last))

} RepeatWhen (identifier relational operators (identifier | Constant))

regular expression of Include

Include identifier (. | Last)

regular expression of Struct

Loli identifier

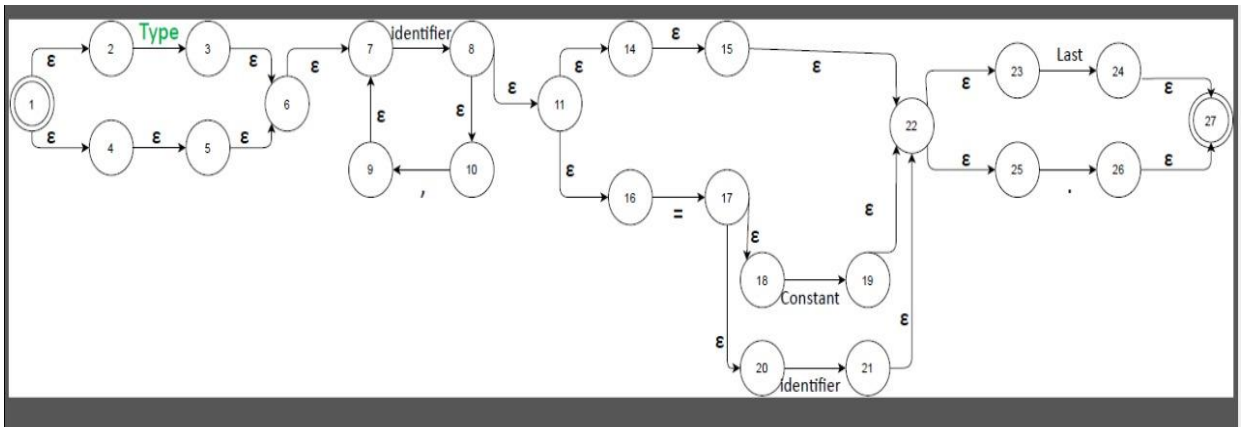
{

(Statement (. | Last))*

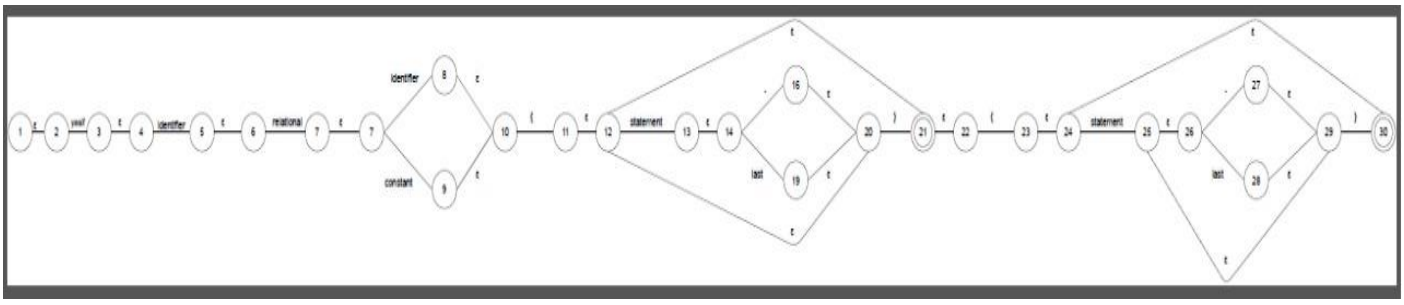
}

Non-deterministic Finite Automata (NFA)

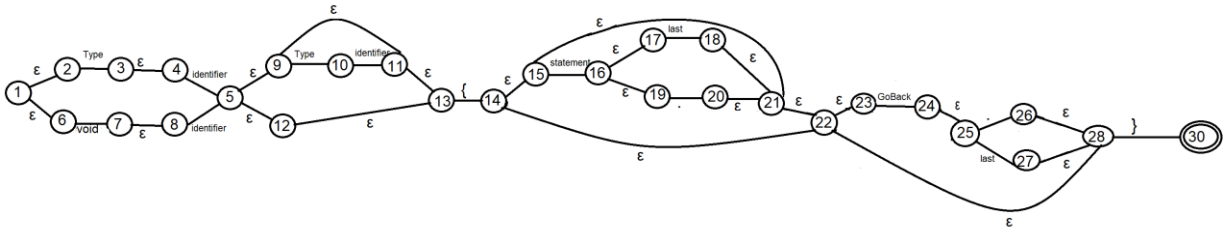
(NFA) of define var



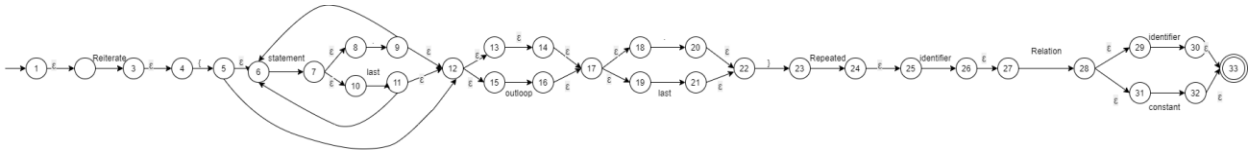
(NFA) of Condition



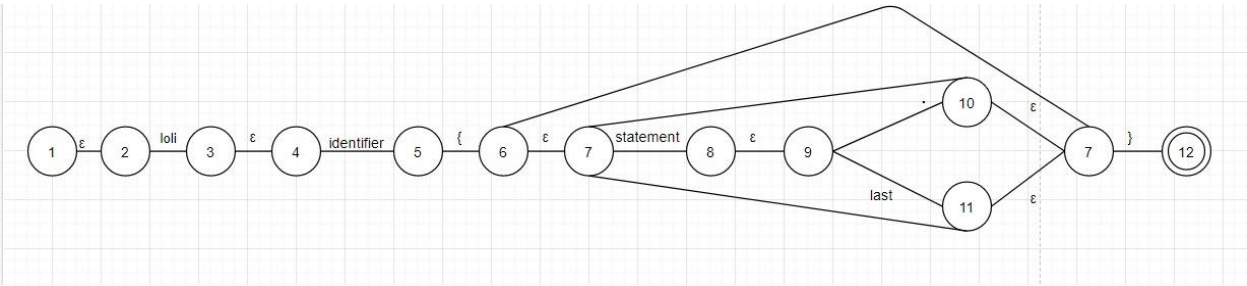
(NFA) of Function



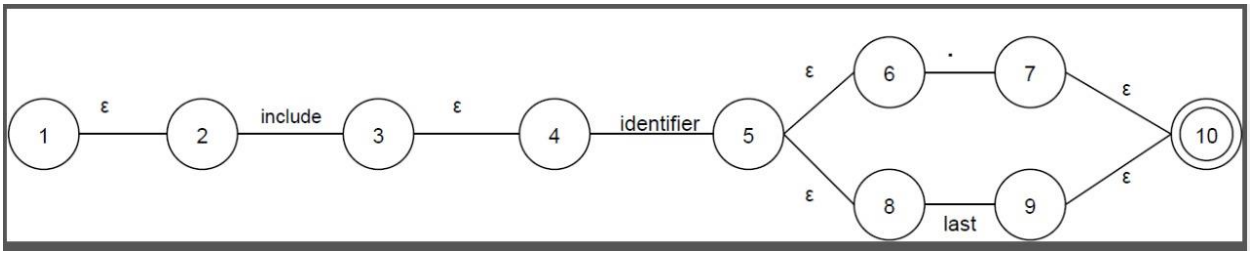
(NFA) of Loop



(NFA) of Struct

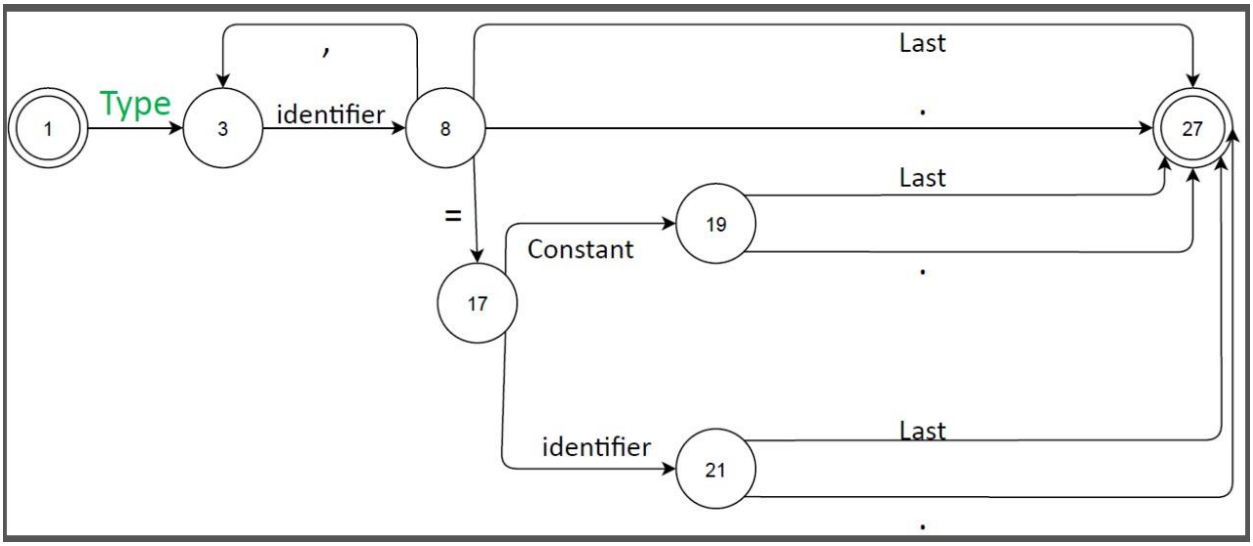


(NFA) of Include

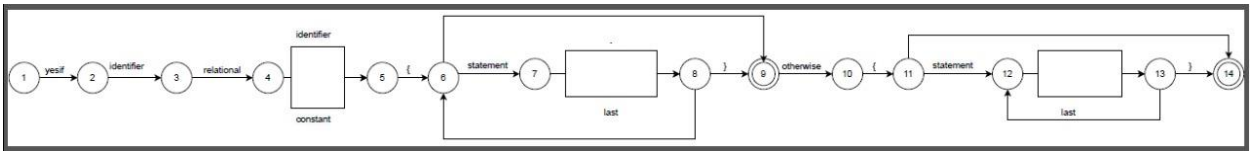


Deterministic Finite Automata (DFA)

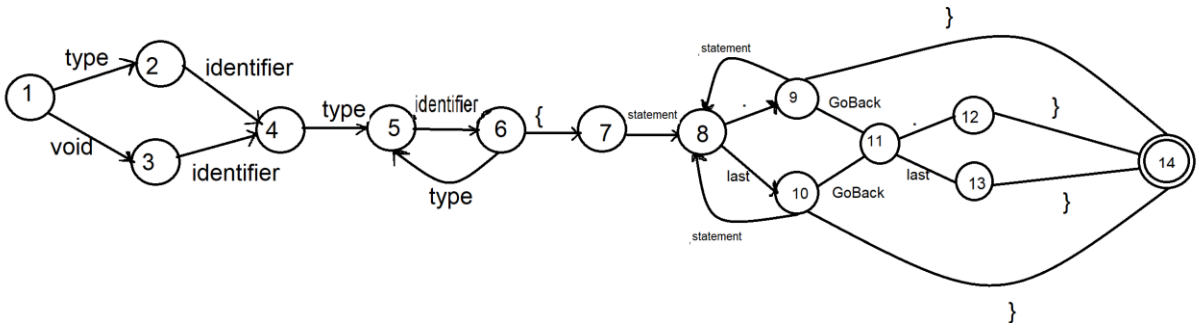
(DFA) of define var



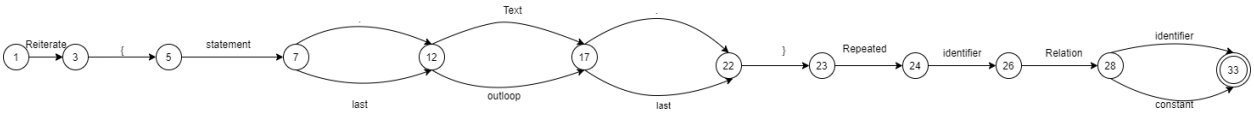
(DFA) of Condition



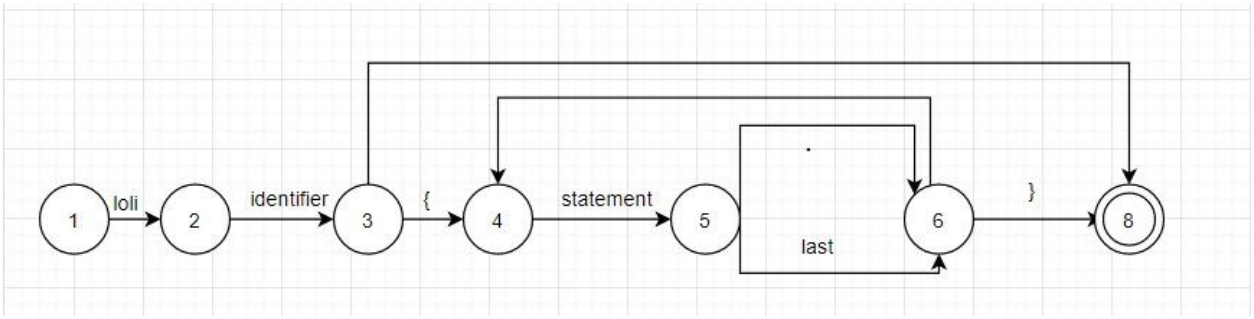
(DFA) of Function



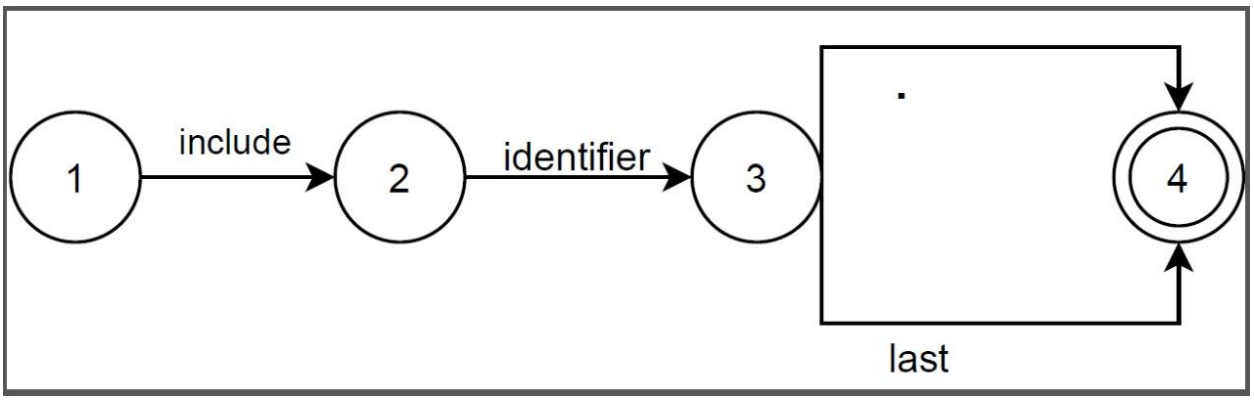
(DFA) of Loop



(DFA) of Struct



(DFA) of Include



Transition Table

Transition Table of Define Variable

state	Type	identifier	constant	,	=	Last	.
1	2						
2		3					
3				2	4	7	7
4		6	5				
5						7	7
6						7	7
7							

Transition Table of Condition

state	identifier	relational	constant	statement	otherwise	yesif	last	.	{	}
1						2				
2	3									
3		4								
4	5		5							
5									6	
6				7						9
7							8	8		
8				7						9
9					10					
10									11	
11				12						14
12							13	13		
13				12						14
14										

Transition Table of Function

	type	void	identifier	{	.	last	statement	GoBack	}	accepting
1	2	3								no
2			4							no
3			4							no
4	5									no
5			6							no
6	5			7						no
7							8			no
8					9	10				no
9								11	14	no
10								11	14	no
11					12	13				no
12									14	no
13									14	no
14										yes

Transition Table of Loop

State	Reiterate	{	Statement	.	Last	ε	outloop	}	Repeation	Identifier	Relation operator	Constant	Accepting
1	3												no
3		5											no
5			7										no
7				12	12								no
12						16	16						no
16				21	21								no
21								22					no
22									23				no
23										25			no
25											27		no
27										32		32	no
32													yes

Transition Table of Struct

state	loli	identifier	statement	last	.	{	}
1	2						
2		3					
3						4	
4			5				
5				6	6		
6			5				8
8							

Transition Table of Include

state	Include	identifier	.	last
1	2			
2		3		
3			4	4
4				