

Identificador

Son las palabras que cumplen el iniciar con una letra y pueden estar seguidas de muchas letras o muchos dígitos.

Creando una posible expresión regular

Alfabeto => { { a-z, A-Z}, { 0-9 } } = { a-z A-Z, 0-9 }

Expresión regular: (a-z A-Z) (a-z A-Z | 0-9)*

Convirtiendo a una gramática regular

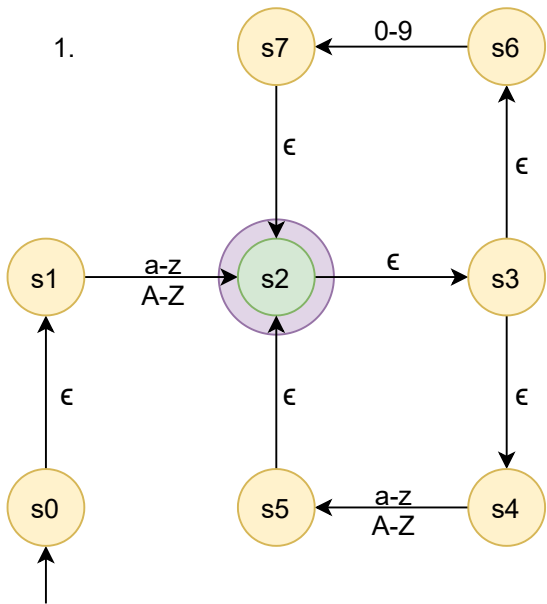
Para mayor compresión: { a-z, A-Z} => \L
{ 0-9 } = \d

$\backslash L (\backslash L \mid \backslash d)^*$ => $S \rightarrow \backslash L$

$\backslash L (\backslash L \mid \backslash d)^*$ => $S \rightarrow \backslash L A$
 $A \rightarrow \backslash L A \mid \epsilon$
 $B \rightarrow \backslash L \mid \backslash d$

Resultado: $S \rightarrow \backslash L A$
 $A \rightarrow BA \mid \epsilon$
 $B \rightarrow \backslash L \mid \backslash d$

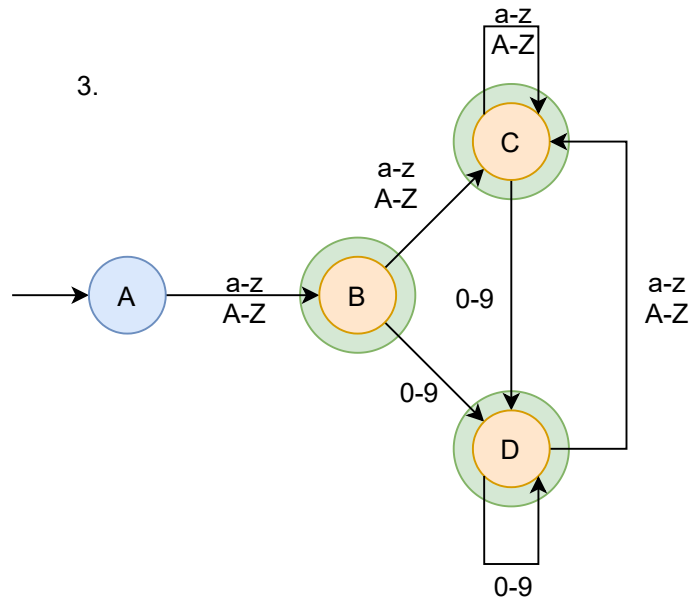
Creando un posible autómatata finito determinista



2.

FT	ε	a-z A-Z	0-9
s0	s1 = A	(A, a-z A-Z) = s2 = B	(A, 0-9) = {}
s2	s3, s4, s6 = B	(B, a-z A-Z) = s5 = C	(B, 0-9) = s7 = D
s5	s2, s3, s4, s6 = C	(C, a-z A-Z) = s5 = C	(C, 0-9) = s7 = D
s7	s2, s3, s4, s6 = D	(C, a-z A-Z) = s5 = C	(C, 0-9) = s7 = D

3.



Reduciendo AFD

1. No hay estados inaccesibles

2.

Estados $\Rightarrow Q = \{ A, B, C, D \}$

Estado Inicial $\Rightarrow A$

Alfabeto $\Rightarrow \Sigma = \{ \{ \{ a-z \}, \{ A-Z \} \}, \{ 0-9 \} \} = \{ a-z A-Z, 0-9 \}$

Aceptación $\Rightarrow B, C, D$

Funciones de Transición \Rightarrow

$$\begin{array}{llll}
 \partial(A, a-z A-Z) = B & \partial(B, a-z A-Z) = C & \partial(C, a-z A-Z) = C & \partial(D, a-z A-Z) = C \\
 \partial(B, 0-9) = D & \partial(C, 0-9) = D & \partial(D, 0-9) = D &
 \end{array}$$

3.

	No Aceptación			Aceptación		
	A			B	C	D
a-z A-Z	B			C	C	C
0-9				D	D	D

4.

	No Aceptación			Aceptación		
	A			B	C	D
a-z A-Z	B			C	C	C
0-9				D	D	D

5.

	No Aceptación			Aceptación		
	s0 = A			s1 = { B, C, D }		
a-z A-Z	B			C	C	C
0-9				D	D	D

6.

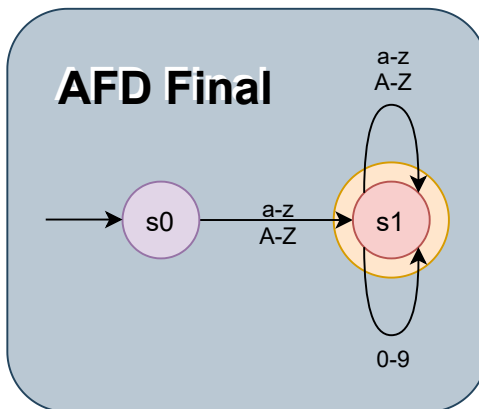
$\partial(s0, a-z A-Z) = s1$	$\partial(s1, a-z A-Z) = s1$
	$\partial(s1, 0-9) = s1$

7.

Estados $\Rightarrow Q = \{ s0, s1 \}$
 Estado Inicial $\Rightarrow s0$
 Alfabeto $\Rightarrow \Sigma = \{ \{ a-z \}, \{ A-Z \} \}, \{ 0-9 \} = \{ a-z A-Z, 0-9 \}$
 Aceptación $\Rightarrow s1$
 Funciones de Transición \Rightarrow

$\partial(s0, a-z A-Z) = s1$

$\partial(s1, a-z A-Z) = s1$
 $\partial(s1, 0-9) = s1$



Número

Son palabras que cumplen con tener al menos un dígito o más, y solo puede contener dígitos.

Creando una posible expresión regular

Alfabeto => { { 0-9 } } = { 0-9 }

Expresión regular: (0-9)+

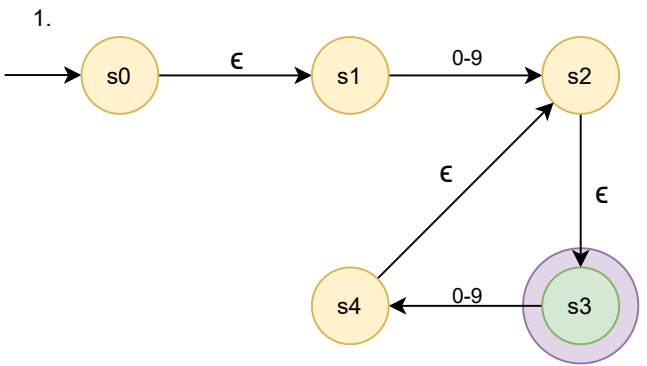
Convirtiendo a una gramática regular

Para mayor compresión: { 0-9 } = \d

\d+ => S -> \d S | \d

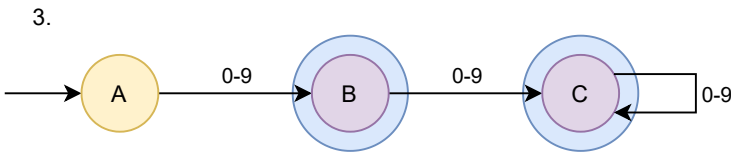
Resultado: S -> \d S | \d

Definiendo Posible AFD



2.

FT	ε	0-9
s0	s1 = A	(A, 0-9) = s2 = B
s2	s3 = B	(B, 0-9) = s4 = C
s4	s2, s3 = C	(C, 0-9) = s4 = C



Reduciendo AFD

1. No hay estados inaccesibles

2.

Estados $\Rightarrow Q = \{ A, B, C \}$

Estado Inicial $\Rightarrow A$

Alfabeto $\Rightarrow \Sigma = \{ \{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 \} \} = \{ 0-9 \}$

Aceptación $\Rightarrow B, C$

Funciones de Transición \Rightarrow

$\partial(A, 0-9) = B$

$\partial(B, 0-9) = C$

$\partial(C, 0-9) = C$

3.

	No Aceptación		Aceptación	
	A		B	C
0-9	B		C	C

4.

	No Aceptación		Aceptación	
	A		B	C
0-9	B		C	C

5.

	No Aceptación		Aceptación	
	s0 = A		s1 = { B, C }	
0-9	B		C	C

6.

$\partial(s0, 0-9) = s1$

$\partial(s1, 0-9) = s1$

7.

Estados $\Rightarrow Q = \{ s0, s1 \}$

Estado Inicial $\Rightarrow s0$

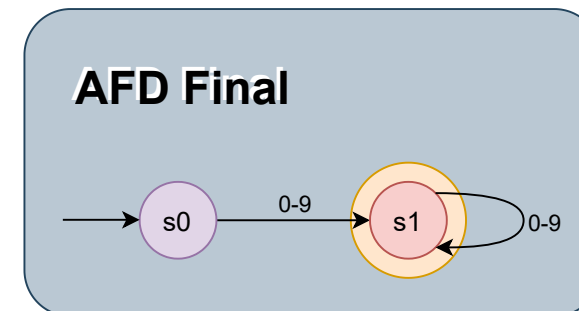
Alfabeto $\Rightarrow \Sigma = \{ \{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 \} \} = \{ 0-9 \}$

Aceptación $\Rightarrow s1$

Funciones de Transición \Rightarrow

$\partial(s0, 0-9) = s1$

$\partial(s1, 0-9) = s1$



Decimal

Son palabras que cumplen con tener al menos un dígito o más, seguido de un punto, seguido de uno o más dígitos.

Creando una posible expresión regular

Alfabeto $\Rightarrow \{ \{ 0-9 \}, \text{punto} \} = \{ 0-9, \text{punto} \}$

Expresión regular: $(0-9)^+ (\text{punto}) (0-9)^+ \Leftrightarrow (0-9)^+ [.] (0-9)^+$

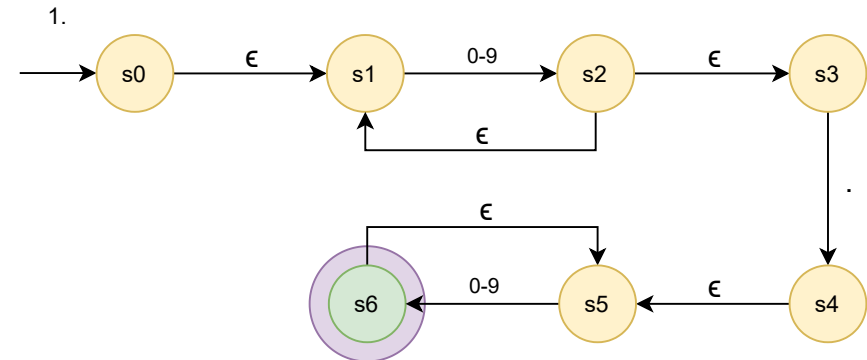
Convirtiendo a una gramática regular

Para mayor compresión: $\{ 0-9 \} = \backslash d ; . = \backslash p$

$(\backslash d)^+ \backslash p (\backslash d)^+ \Rightarrow S \rightarrow$
 $A \rightarrow \backslash d A \mid \backslash d$
 $(\backslash d)^+ \backslash p (\backslash d)^+ \Rightarrow S \rightarrow A \backslash p A$
 $A \rightarrow \backslash d A \mid \backslash d$

Resultado: $S \rightarrow A \backslash p A \Leftrightarrow A . A$
 $A \rightarrow \backslash d A \mid \backslash d$

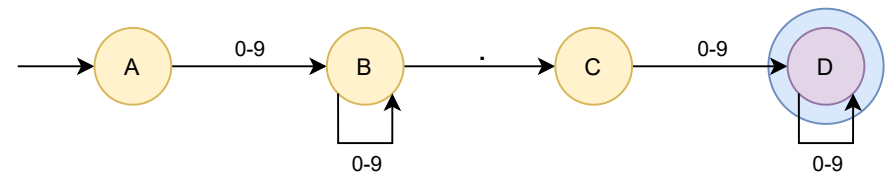
Definiendo Posible AFD



2.

FT	ε	0-9	.
s0	s1 = A	(A, 0-9) = s2 = B	(A, .) = {}
s2	s1, s3 = B	(B, 0-9) = s2 = B	(B, .) = s4 = C
s4	s5 = C	(C, 0-9) = s6 = D	(C, .) = {}
s6	s5 = D	(D, 0-9) = s6 = D	(D, 0-9) = {}

3.



Reduciendo AFD

1. No hay estados inaccesibles

3.

		No Aceptación			Aceptacion	
		A	B	C		D
0-9	B	B	D			D
.		C				

2.

Estados => $Q = \{ A, B, C, D \}$

Estado Inicial => A

Alfabeto => $\Sigma = \{ \{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 \}, \text{punto} \} = \{ 0-9, \text{punto} \}$

Aceptación => D

Funciones de Transición =>

$$\partial(A, 0-9) = B$$

$$\partial(B, 0-9) = B$$

$$\partial(C, 0-9) = D$$

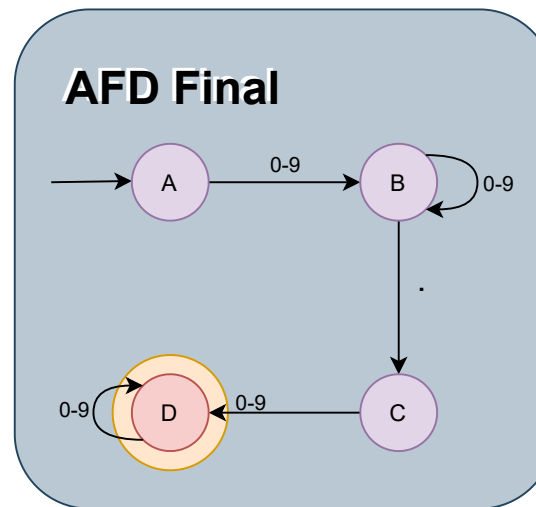
$$\partial(D, 0-9) = D$$

$$\partial(B, \cdot) = C$$

4.

		No Aceptación			Aceptacion	
		A	B	C		D
0-9	B	B	D			D
.		C				

No se puede reducir más el AFD, por lo cual, se tomara el anterior resultado como final.



Puntuación

Ser alguno de los signos de puntuación.

Creando una posible expresión regular

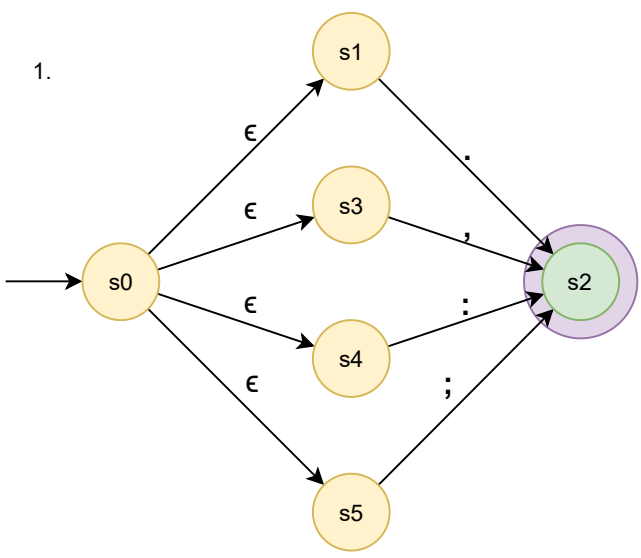
Alfabeto => { [., [,], [:, [;] }

Expresión regular: [.] | [,] | [:] | [;]

Convirtiendo a una gramática regular

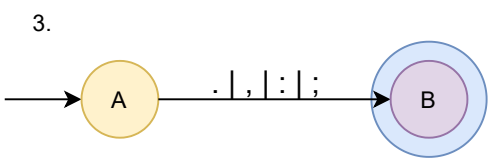
Resultado [.] | [,] | [:] | [;] => S -> A | B | C | D
A -> .
B -> ,
C -> :
D -> ;

Definiendo Posible AFD



2.

FT	ε	.	,	:	;
s0	s1, s3, s4, s5 = A	(A, .) = s2 = B	(A, ,) = s2 = B	(A, :) = s2 = B	(A, ;) = s2 = B
s2	-				



Reduciendo AFD

1. No hay estados inaccesibles

3.

	No Aceptación		Aceptacion	
	A			B
.	B			
,	B			
:	B			
;	B			

2.

Estados => $Q = \{ A, B \}$

Estado Inicial => A

Alfabeto => $\Sigma = \{ [., [,], [:, [:] \}$

Aceptación => B

Funciones de Transición =>

$\partial(A, .) = B$

$\partial(A, ,) = B$

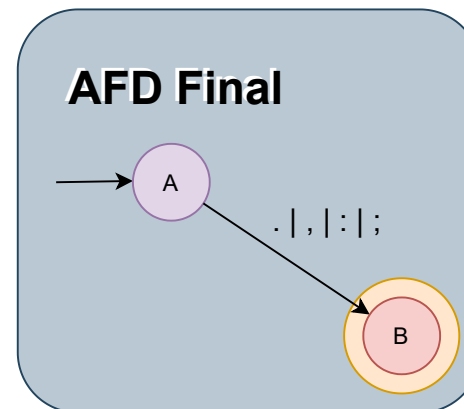
$\partial(A, :) = B$

$\partial(A, ;) = B$

No se puede reducir más el AFD, por lo cual, se tomara el anterior resultado como final.

4.

	No Aceptación		Aceptacion	
	A			B
.	B			
,	B			
:	B			
;	B			



Definiendo Posible AFD

Operador

Ser alguno de los operadores aritméticos.

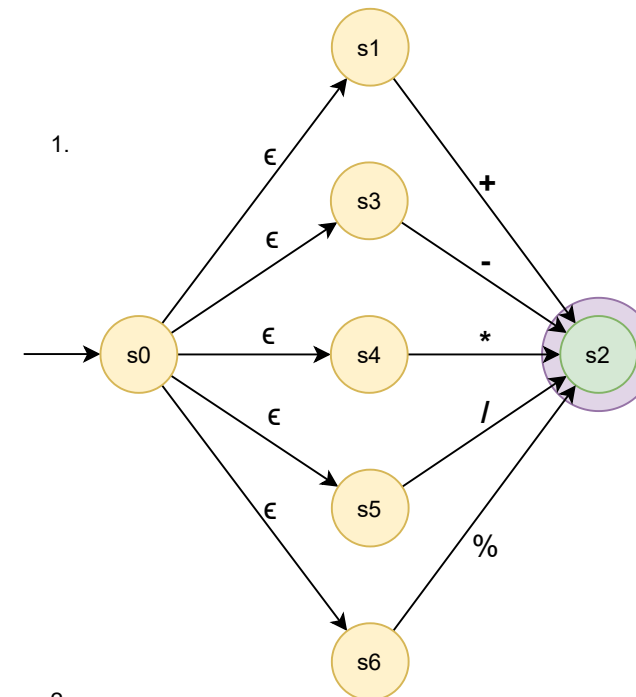
Creando una posible expresión regular

Alfabeto $\Rightarrow \{ +, -, *, /, \% \}$

Expresión regular: $[+] | [-] | [*] | [/] | [\%]$

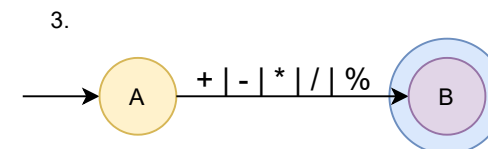
Convirtiendo a una gramática regular

Resultado $[+] | [-] | [*] | [/] | [\%]$ \Rightarrow $S \rightarrow A | B | C | D | E$
 $A \rightarrow +$
 $B \rightarrow -$
 $C \rightarrow *$
 $D \rightarrow /$
 $E \rightarrow \%$



2.

FT	ϵ	+	-	*	/	%
s0	s1, s3, s4, s5, s6 = A	(A, +) = s2 = B	(A, -) = s2 = B	(A, *) = s2 = B	(A, /) = s2 = B	(A, %) = s2 = B
s2	-					



Reduciendo AFD

1. No hay estados inaccesibles

3.

	No Aceptación		Aceptación	
	A			B
+	B			
-	B			
*	B			
/	B			
%	B			

2.

Estados $\Rightarrow Q = \{ A, B \}$

Estado Inicial $\Rightarrow A$

Alfabeto $\Rightarrow \Sigma = \{ +, -, *, /, \% \}$

Aceptación $\Rightarrow B$

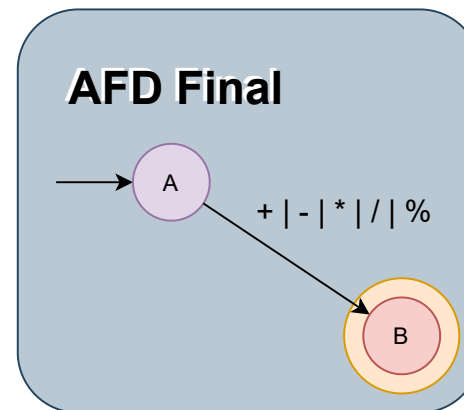
Funciones de Transición \Rightarrow

$$\partial(A, +) = B \quad \partial(A, -) = B \quad \partial(A, *) = B \quad \partial(A, /) = B \quad \partial(A, \%) = B$$

No se puede reducir más el AFD, por lo cual, se tomara el anterior resultado como final.

4.

	No Aceptación		Aceptación	
	A			B
+	B			
-	B			
*	B			
/	B			
%	B			



Agrupación

Ser alguno de los signos de agrupación.

Creando una posible expresión regular

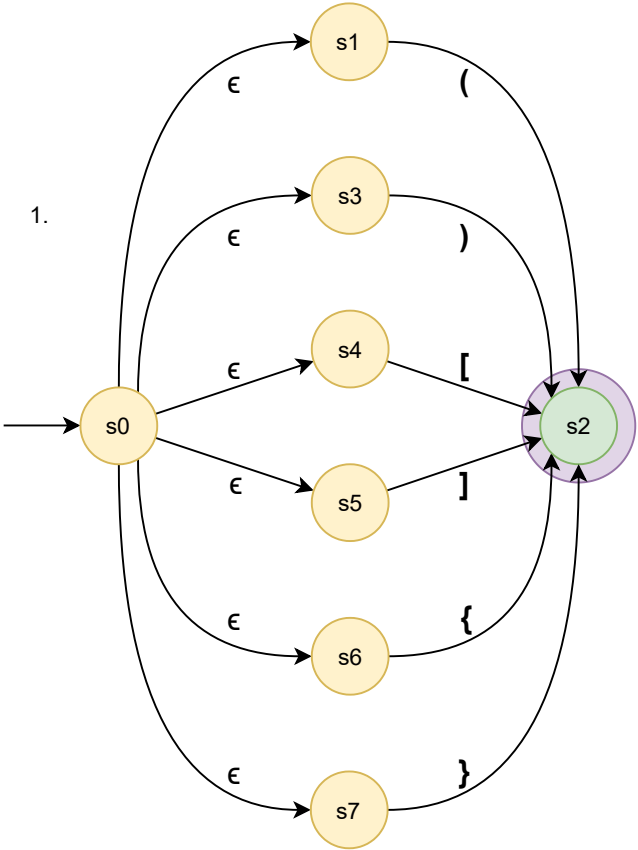
Alfabeto => { (,), [,], {, } }

Expresión regular: [() | [] | { }]

Convirtiendo a una gramática regular

Resultado [() | [] | { }] => S -> A | B | C | D | E | F
A -> (
B ->)
C -> [
D ->]
E -> {
F -> }

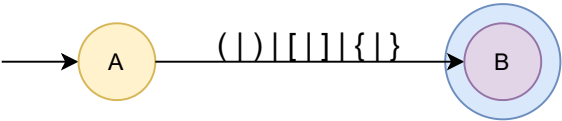
Definiendo Posible AFD



2.

FT	ε	()	[]	{	}
s0	s1, s3, s4, s5, s6, s7 = A	(A, () = s2 = B	(A,)) = s2 = B	(A, [] = s2 = B	(A,]) = s2 = B	(A, { } = s2 = B	(A, }) = s2 = B
s2	-						

3.



Reduciendo AFD

1. No hay estados inaccesibles

3.

	No Aceptación		Aceptación	
	A			B
(B			
)	B			
[B			
]	B			
{	B			
}	B			

2.

Estados $\Rightarrow Q = \{ A, B \}$

Estado Inicial $\Rightarrow A$

Alfabeto $\Rightarrow \Sigma = \{ (,), [,], \{, \} \}$

Aceptación $\Rightarrow B$

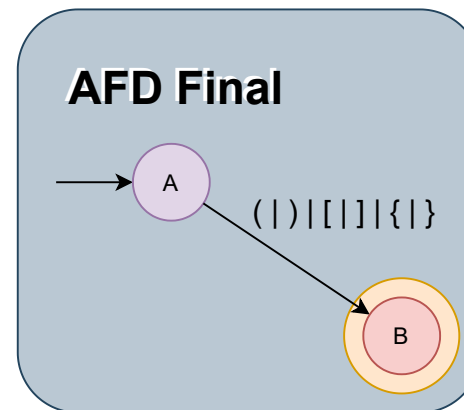
Funciones de Transición \Rightarrow

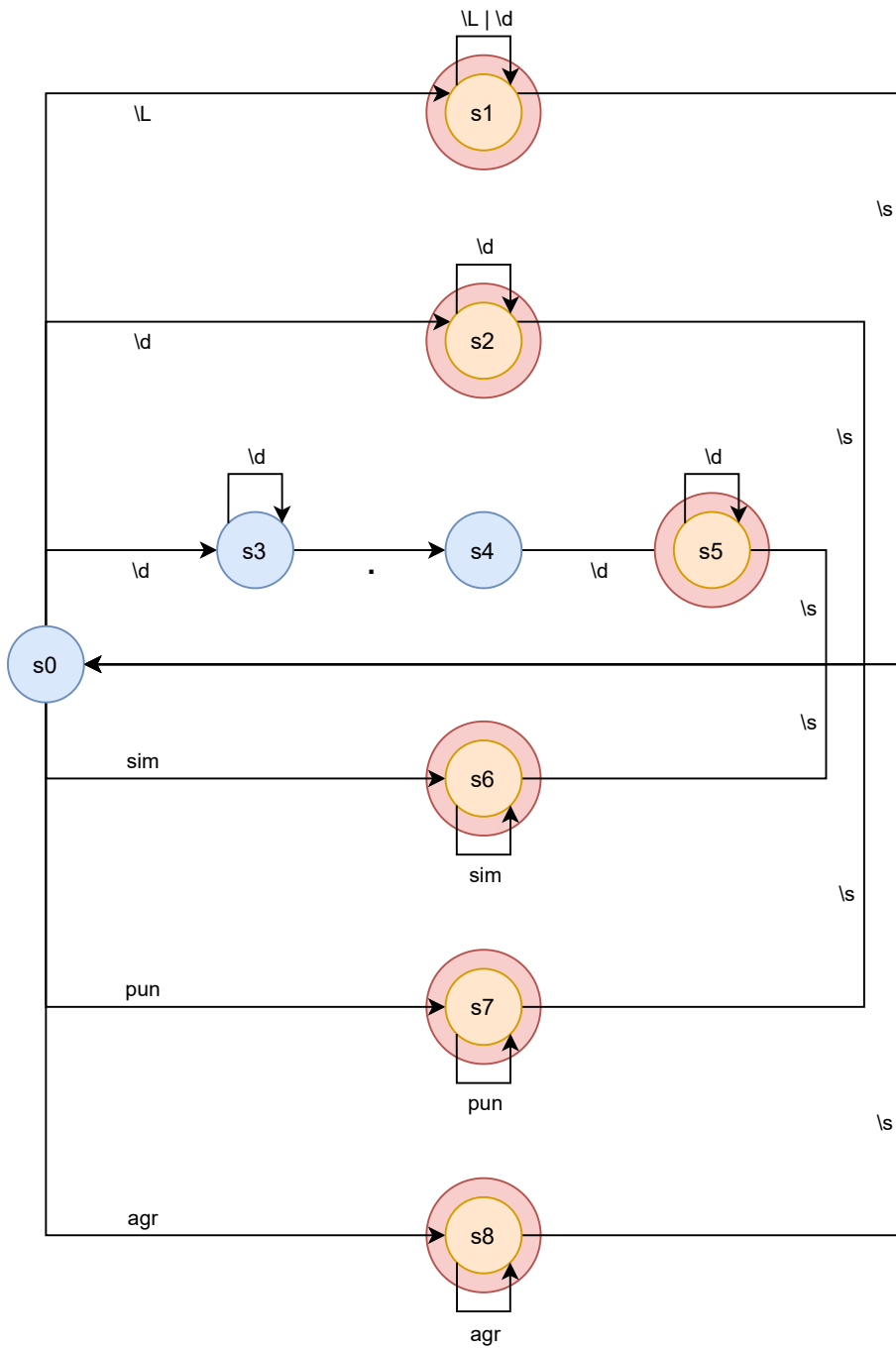
$$\delta(A, () = B \quad \delta(A,)) = B \quad \delta(A, []) = B \quad \delta(A, []) = B \quad \delta(A, \{\} = B \quad \delta(A, \{\} = B$$

No se puede reducir más el AFD, por lo cual, se tomara el anterior resultado como final.

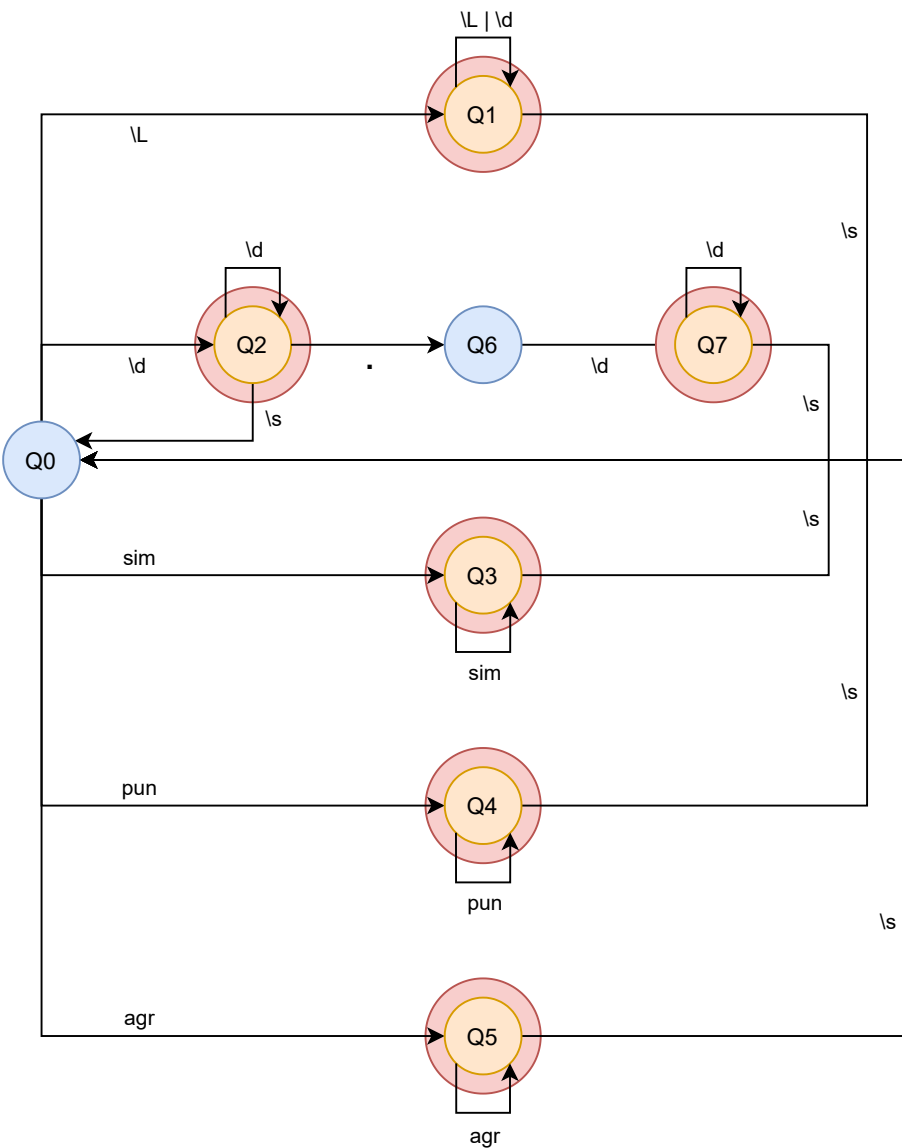
4.

	No Aceptación		Aceptación	
	A			B
(B			
)	B			
[B			
]	B			
{	B			
}	B			





Estado	\L	\d	.	pun	ope	agr	\s
s0 = Q0	s1 = Q1	s2, s3 = Q2	s6 = Q3	s6 = Q3	s7 = Q4	s8 = Q5	
Q1	s1 = Q1	s1 = Q1					s0 = Q0
Q2		s2, s3 = Q2	s4 = Q6				s0 = Q0
Q3			s6 = Q3	s6 = Q3			s0 = Q0
Q4					s7 = Q4		s0 = Q0
Q5						s8 = Q5	s0 = Q0
Q6		s5 = Q7					
Q7		s5 = Q7					s0 = Q0



Definición Formal

Estados $\Rightarrow Q = \{ Q0, Q1, Q2, Q4, Q5, Q6, Q7 \}$

Estado Inicial $\Rightarrow Q0$

Alfabeto $\Rightarrow \Sigma = \{ a-z A-z, 0-9, ., \{ [., [., [:], [:] \}, \{ +, -, *, /, \% \}, \{ (,), [,], \{, \} \} \}$

Aceptación $\Rightarrow B, C, D$

Funciones de Transición \Rightarrow

$\partial(Q0, a-z A-Z) = Q1$

$\partial(Q0, 0-9) = Q2$

$\partial(Q6, 0-9) = Q7$

$\partial(Q0, .) = Q3$

$\partial(Q0, \text{pun}) = Q3$

$\partial(Q0, \text{ope}) = Q4$

$\partial(Q0, \text{agr}) = Q5$

$\partial(Q1, \backslash s) = Q0$

$\partial(Q4, \backslash s) = Q0$

$\partial(Q1, a-z A-Z) = Q1$

$\partial(Q1, 0-9) = Q1$

$\partial(Q7, 0-9) = Q7$

$\partial(Q2, .) = Q6$

$\partial(Q3, \text{pun}) = Q1$

$\partial(Q4, \text{ope}) = Q4$

$\partial(Q5, \text{agr}) = Q5$

$\partial(Q2, \backslash s) = Q0$

$\partial(Q5, \backslash s) = Q0$

$\partial(Q2, 0-9) = Q2$

$\partial(Q3, .) = Q3$

$\partial(Q3, \backslash s) = Q0$

$\partial(Q7, \backslash s) = Q0$

		\L	\d	.	P	O	A	\s	Alfabeto
-	0	1	2	3	3	4	5	0	
ID	1	1	1					0	
NU	2		2	6				0	
PU	3			3	3			0	
OP	4					4		0	
AG	5						5	0	
-	6		7						
DE	7		7					0	
Token	Estados								

Abr		Conjunto de...
\L	⇒	Letras
\d	⇒	Numeros
.	⇒	Punto
P	⇒	Puntuacion
O	⇒	Operador
A	⇒	Agrupacion
\s	⇒	Espacios

Abr		Token de...
ID	⇒	Identificador
NU	⇒	Numero
PU	⇒	Puntuacion
OP	⇒	Operador
AG	⇒	Agrupacion
DE	⇒	Decimal
-	⇒	Error

