

DFA Specifications

1 Identifier DFA

Rule:

- Start with uppercase letter
- Followed by lowercase / digit / underscore
- Max length 31 (checked separately)

States:

- q_0 = Start
- q_1 = Valid identifier
- q_E = Error

Transition Table:

Current State	Input	Next State
q_0	A–Z	q_1
q_0	other	q_E
q_1	a–z	q_1
q_1	0–9	q_1
q_1	–	q_1
q_1	A–Z	q_E
q_1	other	Accept

Accepting state: q_1

2 Integer DFA

Rule:

- One or more digits

States:

- q_0 = Start
- q_1 = Digit seen
- q_E = Error

Transition Table:

Current State	Input	Next State
q_0	0–9	q_1

Current State	Input	Next State
q0	other	qE
q1	0–9	q1
q1	.	Float DFA
q1	other	Accept

Accepting state: q1

3 Float DFA

Rule:

- digits '.' digits
- max 6 digits after decimal
- optional exponent

States:

- q0 = Start
- q1 = integer part
- q2 = dot seen
- q3 = decimal digits
- q4 = exponent symbol
- q5 = exponent sign
- q6 = exponent digits
- qE = Error

Transition Table:

State	Input	Next
q0	0–9	q1
q1	0–9	q1
q1	.	q2
q2	0–9	q3
q3	0–9 (≤ 6 digits)	q3
q3	e/E	q4
q4	+/-	q5
q4	0–9	q6
q5	0–9	q6

State	Input	Next
q6	0–9	q6
q3/q6	other	Accept

Accepting states: q3, q6

4 Boolean DFA

Recognizes: true, false

For "true"

State	Input	Next
q0	t	q1
q1	r	q2
q2	u	q3
q3	e	q4
q4	other	Accept

For "false"

State	Input	Next
q0	f	q1
q1	a	q2
q2	l	q3
q3	s	q4
q4	e	q5
q5	other	Accept

Accepting states: q4 (true), q5 (false)

5 String DFA

Rule:

- Enclosed in double quotes
- Allows escape sequences

States:

- q0 = Start

- q1 = Inside string
- q2 = Escape
- q3 = Closing quote
- qE = Error

Transition Table:

State	Input	Next
q0	"	q1
q1	\	q2
q1	"	q3
q1	other	q1
q2	any char	q1

Accepting state: q3

6 Character DFA

Rule:

- Single quotes
- One char or valid escape
- Max 1 character inside

States:

- q0 = Start
- q1 = Opening '
- q2 = Character read
- q3 = Closing '
- qE = Error

Transition Table:

State	Input	Next
q0	'	q1
q1	\	q2
q1	any single char	q2
q2	'	q3
q2	other	qE

Accepting state: q3

7 Comment DFA

Single Line Comment: ##

State	Input	Next
q0	#	q1
q1	#	q2
q2	not newline	q2
q2	newline	Accept

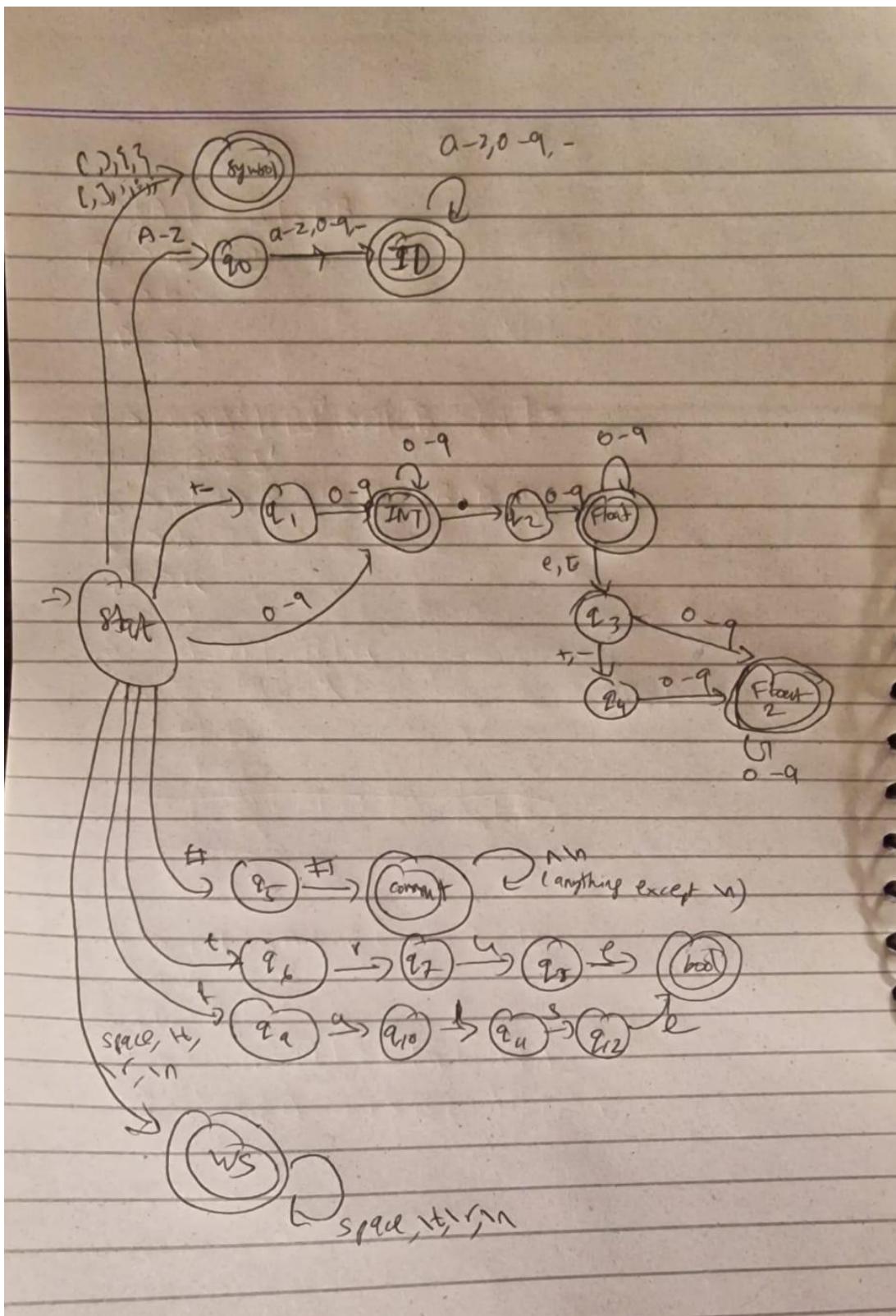
Accepting state: q2 (on newline)

Multi-line Comment: #| |#

State	Input	Next
q0	#	q1
q1		q2
q2	other	q2
q2		q3
q3	#	q4
q3	other	q2

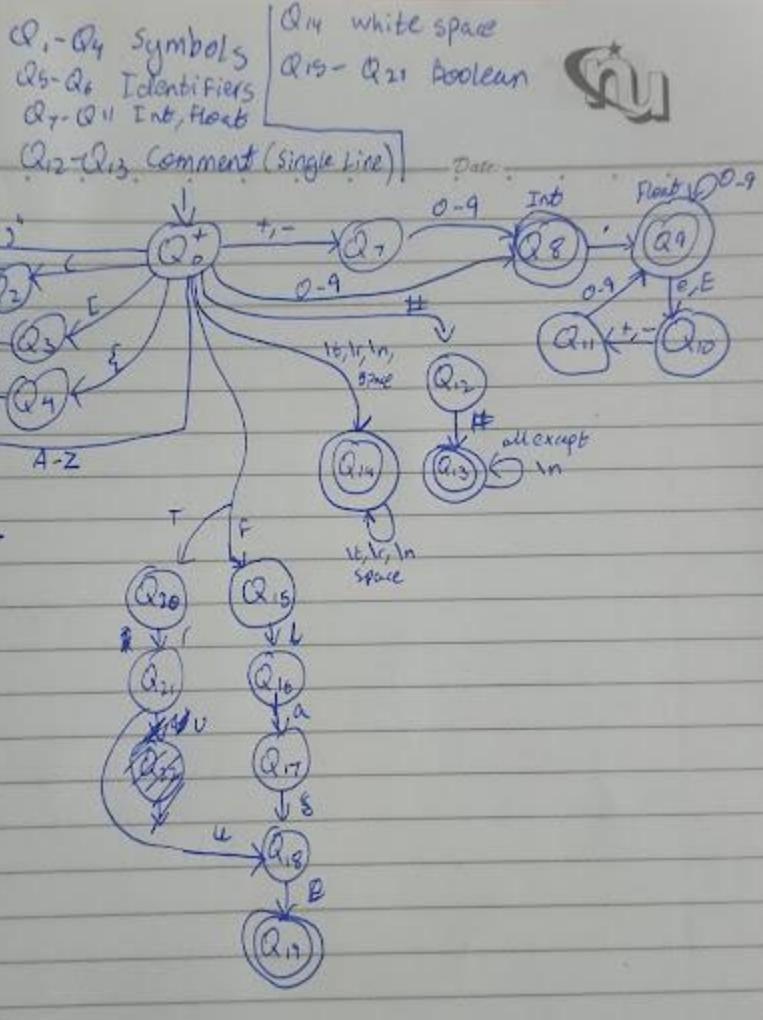
Accepting state: q4

DFA-Simple

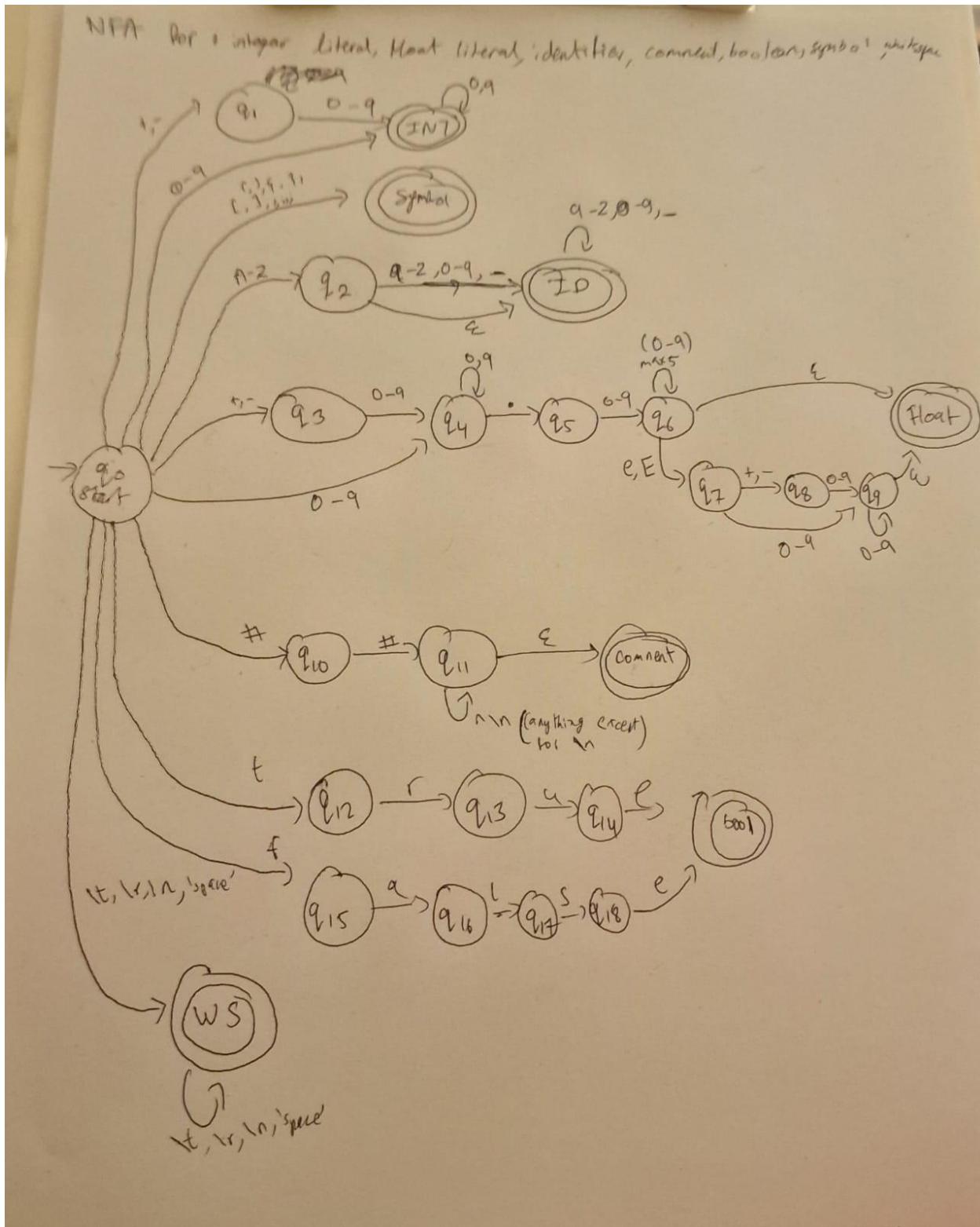


DFA-min states:

FAST



NFA:



Regex:

1.

Keywords

- **Regex:**
(start|finish|loop|condition|declare|output|input|function|return|break|continue|else
)
- *Note:* Case-sensitive.

2.

Identifiers

- **Regex:** [A-Z][a-z0-9_]{0,30}
- *Rule:* Must start with an uppercase letter, followed by lowercase letters, digits, or underscores. Max length 31.

3.

Integer Literals

- **Regex:** [+]?[0-9]+
- *Examples:* 42, +100, -567.

4.

Floating-Point Literals

- **Regex:** [+]?[0-9]+\.[0-9]{1,6}([eE][+-]?[0-9]+)?
- *Rule:* Optional sign, required digits before dot, required digits (max 6) after dot, optional exponent.

5.

String Literals

- **Regex:** \\$"([^\\n]|\\(["\\ntr]))*\\$\n
- *Rule:* Starts/ends with \$. Supports escape sequences \", \\, \n, \t, \r.
- *Note:* Your implementation code used ", but the PDF explicitly specifies \$. Use this regex for the design document to match the spec.

6.

Character Literals

- **Regex:** '([^\\n]|\\(["\\ntr]))'
- *Rule:* Enclosed in single quotes.

7.

Boolean Literals

- **Regex:** (true|false)

- Case-sensitive.

8.

Operators

- **Arithmetic:** ($+|-|^{*}|/|^{*}\%$)
- **Relational:** ($==|!=|<=|>=|<|>$)
- **Logical:** ($\&\&|\vee\vee|!$)
- **Assignment:** ($=|+=|-=|^{*}=|\vee=$)
- **Inc/Dec:** ($+|+|--$)
- **Combined Regex:** ($^{*}^{*}==|!=|<=|>=|\&\&|\vee\vee|\vee+=|\vee-=|\vee^{*}=|\vee=|\vee+|--|[+|-^{*}/%<>=!]$)

9.

Punctuators

- **Regex:** [(){}[\]]`,::`

10.

Comments

- **Single-line:** ##[^
n]*
- **Multi-line:** #|(.)|s)*?|#
- *Rule:* Single line starts with ##. Multi-line is enclosed in #| ... |#.

11.

Whitespace

+4

- **Regex:** [\t\r\n]+