# PATTERN — TOKEN

| PATTERN | TOKEN |
|---------|-------|
| integer | INTEGER |
| real | REAL |
| of | OF |
| boolean | BOOLEAN |
| array | ARRAY |
| start | START |
| end | END |
| declare | DECLARE |
| module | MODULE |
| driver | DRIVER |
| program | PROGRAM |
| get_value | GET_VALUE |
| print | PRINT |
| use | USE |
| with | WITH |
| parameters | PARAMETERS |
| takes | TAKES |
| input | INPUT |
| returns | RETURNS |
| for | FOR |
| in | IN |
| switch | SWITCH |
| case | CASE |
| break | BREAK |
| default | DEFAULT |
| while | WHILE |
| AND | AND |
| OR | OR |
| true | TRUE |
| false | FALSE |

KEYWORDS / LOOK-UP TABLE [26+4]

**ACCEPT STATE OF ID**

Check the above LOOK-UP TABLE for a match. The lexemes with the above patterns are tokenized with corresponding token names.

If they do not match with any pattern, they are tokenized as ID.

Send control back to START.

Encapsulate LEXEMES, LINE NUMBER and TOKEN.

---

Transition diagram states and labels:

[/] → DIV
[-] → MINUS
[+] → PLUS
[<] → LT
  [<] → DEF, [=] → LE
[=] → (state) [=] → EQ
[<] → DEF → DRIVERDEF
ASSIGNOP
[=] → (state)
[:] → COLON
[/] → ASSIGNOP

[0-9] → RNUM, [0-9], [+/-], [E/e]
[0-9] → NUM, [.] , [0-9] → RNUM, LOOKAHEAD, [.]

[a-z|A-Z|_] → ID, [a-z|A-Z|_|0-9]

[\n] → #
[/] , [\t] → %

[=] → NE, [!]
[.] → RANGEOP, [.]

OTHER, OTHER
[*] → MUL, [*]
[\n], [\n], OTHER
[*] → @, [\n]
$

GE, [=]
[>] → GT, [>] → ENDDEF → DRIVERENDDEF
[>]

BC [)]
BO [(]
SQBC []]
SQBO [[]
COMMA [,]
SEMICOL [;]

S (START)

## ACCEPT STATE ⊚

Control back to START

Encapsulate LEXEMES, LINE NUMBERS and/or NUMERIC VALUE along with the TOKEN

↦ NUMBERS | NUM, RNUM

TOKENIZE and find NUMERIC equivalent value.

↦ KEYWORDS & IDENTIFIERS

Check LOOKUP TABLE and tokenize it accordingly. Explained in detail under the lookup table.

↦ OPERATORS & SPECIAL SYMBOLS

TOKENIZE according to their respective token.

↦ WHITE SPACES & COMMENTS

(#) : NEWLINE, Increment line number

(%) : TB/SPACES, Ignore

(@) : COMMENT, Increment line number NEWLINE → [INTERMEDIATE STATE]

($) : COMMENTEND, Ignore

## TRAP STATE ----- (IMPLICIT)

Report ERROR & send control back to START.

All the UNSPECIFIED transitions go to the TRAP.

It reports LEXICAL ERROR with LINE NUMBER.

## INTERMEDIATE STATE ◯, ⊗

Wait for the input and move to NEXT state.

Move to TRAP if no valid transition for the input.

RETRACTION if required, from ⊗

## RETRACTION STATE □

Retract the FORWARD pointer by a state and TOKENIZE the LEXEMES, LINE NUMBER and/or NUMERIC VALUE upto the previous accept state.

Control back to START

## START STATE Ⓢ

Wait for the input and move to NEXT state

Move to TRAP if no valid transition for the input

2020A7PS1691P    ANTRIKSH  SHARMA  GROUP :
2020A7PS0313P    HRISHIKESH  HARSH    NUMBER
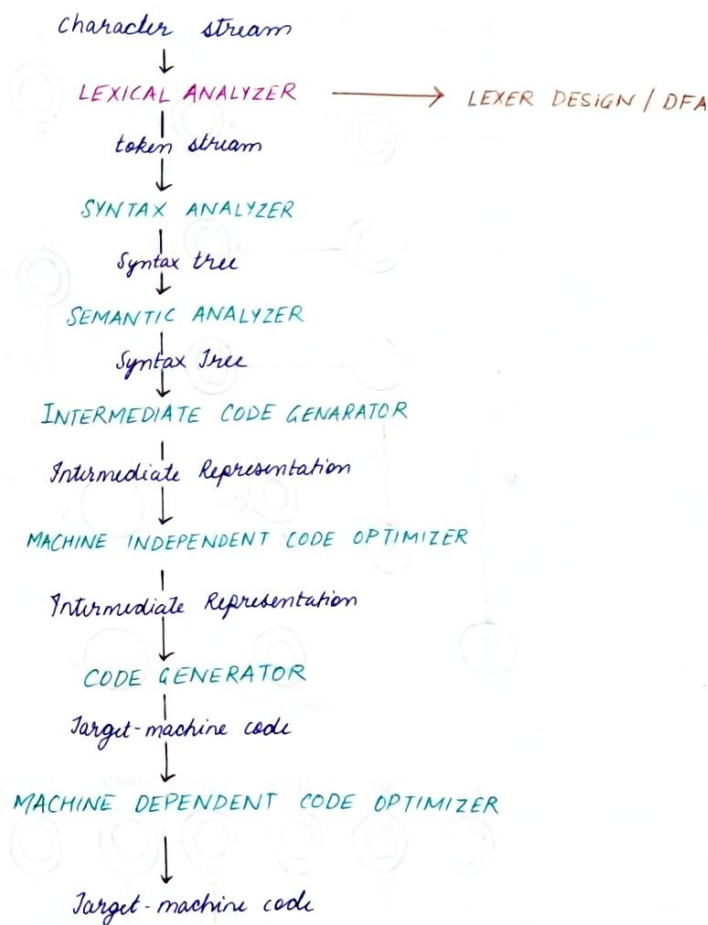2020A7PS0013P    KAUSTAB  CHOUDHURY
2020A7PS0078P    SHASHANK  SHREEDHAR  BHATT
2020A7PS0110P    HARSH  PRIYADARSHI

## LEXER DESIGN -- DFA CONSTRUCTION

character stream
↓
LEXICAL ANALYZER ⟶ LEXER DESIGN/DFA
↓
token stream
↓
SYNTAX ANALYZER
↓
Syntax tree
↓
SEMANTIC ANALYZER
↓
Syntax Tree
↓
INTERMEDIATE CODE GENERATOR
↓
Intermediate Representation
↓
MACHINE INDEPENDENT CODE OPTIMIZER
↓
Intermediate Representation
↓
CODE GENERATOR
↓
Target-machine code
↓
MACHINE DEPENDENT CODE OPTIMIZER
↓
Target-machine code

PHASES OF A COMPILER [TB]