1 → 8th jan
2 → 15th jan
3 → 29th jan
4 → 5th feb
~~5 → 12th feb~~

Practical No. - 1.

Y.C.C.E
DATE :

Aim :- To study lex tool.

Theory :-

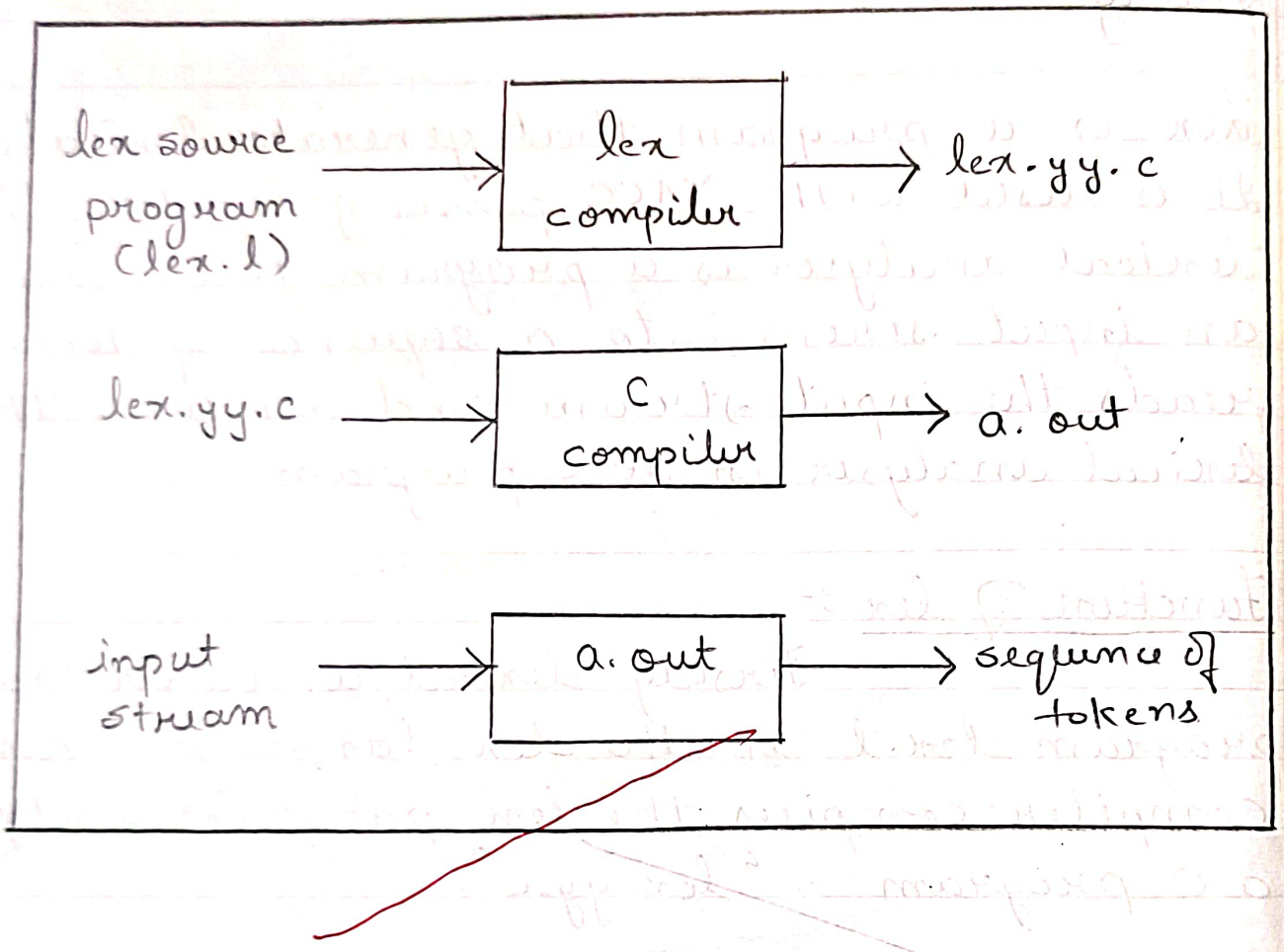
Lex is a program that generate lexical analyses. It is used with YACC passer generator. The lexical analyser is a program that transforms an input strem into a sequence of tokens. It reads the input stream and produces the lexical analyser in the C program.

Function of lex :-

Firstly lexical analyser creates a program lex.l in the lex language. Then lex compiler compiles the lex program and provides a C program - 'lex.yy.c'.

Finally the C compiler compiles the lex.yy.c and converts it into a.out file.

The a.out is the lexical analyser that transform an input stream into a sequence of tokens.


lex source program (lex.l)
lex compiler
lex.yy.c
lex.yy.c
C compiler
a.out
input stream
a.out
sequence of tokens

Y.C.C.E
DATE :

lex File Format :-

{ definitions / declaration }
% %
{ transaction rules }
% %
{ user subroutines / auxilary procedures }

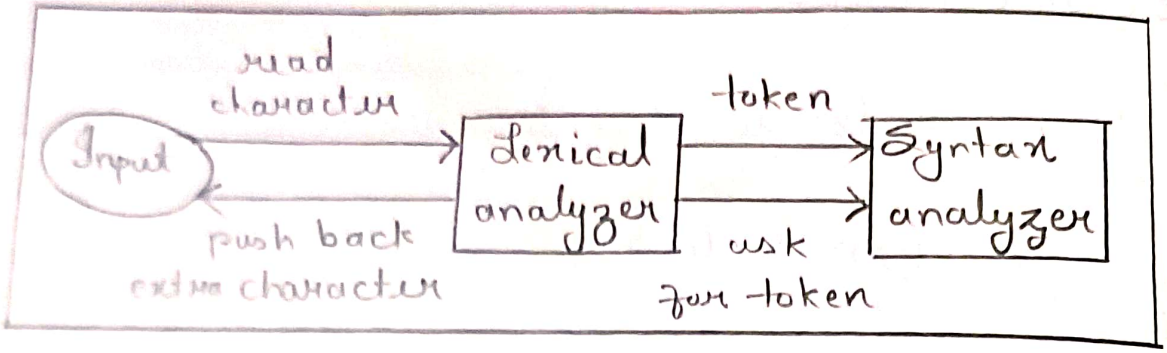
A lex program into 3 sections by % % delimeters

Declaration Section includes declarations of variables, manifeis, constants and regular definition.

Transaction Rules : Transaction rules for lex program are of the form →

P1	{action 1}
P2	{action 2}
P3	{action 3}
....	.....
....	.....

When each P is a regular expression and each action is a program fragment describing what action, the lexical analyser should take when a pattern matches a lexeme


read
character
Input
Lexical
analyzer
token
Syntax
analyzer
push back
extra character
ask
for token

Y.C.C.E
DATE :

User subroutines are auxillary procedures needed by the actions. The subroutines can be loaded with lexical analyser and compiled seperately.

Lexical analysis :-

Lexical analysis is the first slip in compilation of a program. It is the lexical analyser who called as scanners, that does the lexical analysis. It converts high level languages source code into a sequence of token.

- Lexical analysis can be implemented with deterministic finite automata.
- The output is a sequence of token that is sent to the parser for syntax analysis.

Token :- A lexical token is a sequence of characters that can be treated as a unit in the Grammar of programming language.
Ex:- Keywords, Identifier, special symbols, constants, operators, strings.

Lexeme :- The sequence of character matched by a pattern to form the corresponding token or a sequence of input character that comprises a single token is called a lexeme.
Eg :- "float", "return", "=", "-", "273", ";", etc.

Y.C.C.E
DATE :

Working of Lexical Analyzer :-

→ **input preprocessing** : This stage involves cleaning up in the input text and preparing it for lexical analysis. This may include removing comments and other non essensial characters from input text.

→ **Tokenization** : Procers of breaking the input text into a sequence of tokens

→ **token classification** : In this stage, the lexer determines the types of each token.
Ex :- in programming language, the lexer might classify keywords, identifies, operations as seperate token types.

→ **token validation** : In this stage, the lexer checks that each token is valid according to the rules of the programming language.

→ **output generations** : In this final stage, the lexer generates the output of the lexical analysis procers which is typically a list of token.

Y.C.C.E
DATE :

Now, consider the program

```
int main()
{
    int a, b;
    a = 10;
    return 0;
}
```

The valid tokens are: 'int', 'main', '(', ')', '{', 'int', 'a', 'b', ';', 'a', '=', '10', ';', 'a', '=', '10', 'return', '0', ';', '}'.

Some common commands used

yywrap() Returns a lex encounters the end of file, it calls yywrap functions. If yywrap return non-zero value yylex terminates and return to main. If programmer wants to scan more than one input file, then yywrap should return 0.

yylex() Returns a value indicating the type of token that has been obtains.

yyin() It points to an input file which is to be scanned.

Y.C.C.E
DATE :

yyout It points to a file where it has to keep the output file.

yytext a buffer that holds the input character that actually match the pattern (lexeme) or say a pointer to the matched string.

yyerror user defined error handling function used in ylex. It is called when syntax error is encountered during parsing.

yylval : variable used to associate attribute or values with token.

yyleng : yyleng in integer variable that holds the length of matched text in yytext.

yylines : It is an integer variable that keeps track of current lineno. in the input file or stream

Conclusion :-

Thus we have successfully executed the lex tool.

18/3/24

13+