Date __/__/____

Assignment - B2

* Title :- Lexical analysis to generate tokens.

* Problem statement :- Write a problem using LEX specifications to implement lexical analysis phase of compiler to generate tokens of subset of Java Program.

* Objective :- Understand the importance and usage of LEX tool.

* Theory :

During the first phase of compiler, it reads the input and converts strings in the source to tokens with regular expressions, we can specify patterns to lex so it can generate code that will allow it to scan & match strings in the input.
Lex has it's limitations. Lex cannot be used to recognize nested structures such as parantheses.

• ⇒

Pattern	Method
.	any character except new line
\n	new line
\t	tab
^	beginning of line
$	end of line.

Table 1 :- Special characters

Date __/__/____

Saathi Notebooks

Pattern	Method
?	zero or one copy of the preceeding exp.
*	zero or more copies of previous exp.
+	one or more copies of previous exp.
a|b	a or b
(ab)+	one or more copies of (ab).
"a+b"	literal "a+b"
abc	abc
abc*	abc, ab, abcc, abccc.....
"abc*"	literal abc*
abc+	abc, abcc, abccc.....
a(bc)+	abc, abcbc, abcbcbc.....
a(b|b)	[illegible]

Table 2: Operators

Pattern	Method
[abc]	one of: a, b, c
[a-z]	any letter a->z.
[a|-z]	one of: a -z
[-az]	one of: -az.
[A-Za-z0-9]+	one or more alphanumeric characters.
[\t\n]+	whitespace
[^ab]	anything except: a b
[a^b]	one of: a ^ b.
[a|b]	one of: a | b.

Table 3: Characters class.

→

Date ___/___/_____

Name	Function
int yylex(void)	call to invoke lex, return token
char *yytext	pointer to matched string.
yyleng	length of matched string.
yylval	value associated with token.
int yywrap(void)	wrapup, return 1 if done, 0 if not done
FILE *yyout	output file
FILE *yyin	input file.
INITIAL	initial start algorithm.
BEGIN	condition switch start condition
ECHO	write matched string.

Table 4:- Lex predefined values.

Regular expressions are used for pattern matching. A character class defines a single character & normal operators lose their meaning. Two operators allowed in a character class are the hyphen ("-") & circumflex ("^"). When used between two characters, the hyphen represents a range of characters. The circumflex, when used as the first character, negates the expression.

⇒ Definitions
%%
Rules
%%
subroutines.

E.g:- %%
/* match everything except new line */
. ECHO;
/* match newline */

Page No.

Date __/__/____

```
In ECHO;
%%
int yywrap (void) {
 return 1;
 }
int main (void) {
    yylex();
    return 0;
    }
```

* Variables of lex program

1) yytext() :- Whenever the scanner matches a token, the text of the token is stored in null terminated string yytext.

2) yylen :- length of yytext.

3) yylex() :- Scanner created by the lex has the entry point yylex().

* Testcases :-

Input	Expected O/P	Actual O/P	status.
1) public static void main (string[] args) { int a = 10; }	public static void main (string[] args) is main function. { , } => Brackets int → integer datatype a → variable = → assignment, 10 → Number	public static void main (string[] args) is main function { , } => Brackets. int → integer datatype a → variable = → assignment, 10 → Number	Pass.

Page No.

Date __/__/____

Saathi Notebooks

* Conclusion:-

Thus, I successfully implemented lexical Analysis to generate tokens for subset of java program.

Page No.