CSP353 Lab

Experiment: 03 (CO-1)

Implement Lexical analyzer using Lex compiler.

Ravikant Nirala
Assistant Professor, CSE Department, Sharda University

Lex and Yacc compiler installation

Installer Required in same directory (Program Files (x86))

1. Flex (https://gnuwin32.sourceforge.net/packages/flex.htm)
2. Bison (https://gnuwin32.sourceforge.net/packages/bison.htm)
3. Dev++ (https://www.bloodshed.net/)

Set Path

C:\Program Files (x86)\GnuWin32\bin;C:\Program Files (x86)\Dev-Cpp\MinGW64\bin

Command to Run

Step 1: flex file.l
Step 2: gcc lex.yy.c
Step 3: .\a.exe

Implement Lexical analyzer using Lex compiler.

```
%{
#include <stdio.h>
%}

%%
[a-zA-Z_][a-zA-Z0-9_]*       { printf("IDENTIFIER: %s\n", yytext); }
[0-9]+                       { printf("NUMBER: %s\n", yytext); }
[+\-*/=]                     { printf("OPERATOR: %s\n", yytext); }
[ \t\n]                      { /* Ignore whitespace */ }
.                            { printf("UNKNOWN TOKEN: %s\n", yytext); }
%%

int main() {
   printf("Enter input (Ctrl+D to end):\n");
   yylex();
   return 0;
}
int yywrap() {
   return 1;
}
```

```
Enter input (Ctrl+D to end):
a = 5+b
IDENTIFIER: a
OPERATOR: =
NUMBER: 5
OPERATOR: +
IDENTIFIER: b
```