**3. Implementation of Lexical Analyser Using LEX Tool**

lex:

%{

int COMMENT=0;

%}

identifier [a-zA-Z][a-zA-Z0-9]\*

%%

#.\* { printf("\n%s is a PREPROCESSOR DIRECTIVE",yytext);}

int |

float |

char |

double |

while |

for |

do |

if |

break |

continue |

void |

switch |

case |

long |

struct |

const |

typedef |

return |

else |

goto {printf("\n\t%s is a KEYWORD",yytext);}

"/\*" {COMMENT = 1;}

"\*/" {COMMENT = 0;}

{identifier}\( {if(!COMMENT)printf("\n\nFUNCTION\n\t%s",yytext);}

\{ {if(!COMMENT) printf("\n BLOCK BEGINS");}

\} {if(!COMMENT) printf("\n BLOCK ENDS");}

{identifier}(\[[0-9]\*\])? {if(!COMMENT) printf("\n %s IDENTIFIER",yytext);}

\".\*\" {if(!COMMENT) printf("\n\t%s is a STRING",yytext);}

[0-9]+ {if(!COMMENT) printf("\n\t%s is a NUMBER",yytext);}

\)(\;)? {if(!COMMENT) printf("\n\t");ECHO;printf("\n");}

\( ECHO;

= {if(!COMMENT)printf("\n\t%s is an ASSIGNMENT OPERATOR",yytext);}

\<= |

\>= |

\< |

== |

\> {if(!COMMENT) printf("\n\t%s is a RELATIONAL OPERATOR",yytext);}

%%

int main(int argc,char \*\*argv)

{

if (argc > 1)

{

FILE \*file;

file = fopen(argv[1],"r");

if(!file)

{

printf("could not open %s \n",argv[1]);

exit(0);

}

yyin = file;

}

yylex();

printf("\n\n");

return 0;

} int yywrap()

{

return 0;

}

second.c

#include <stdio.h>

int main()

{

int number;

printf("Enter an integer: ");

scanf("%d", &number);

if(number % 2 == 0)

printf("%d is even.", number);

else

printf("%d is odd.", number);

return 0;

}

**Output:**

C:\Users\16cseb47\Desktop\pcd>flex first.l

C:\Users\16cseb47\Desktop\pcd>gcc lex.yy.c

C:\Users\16cseb47\Desktop\pcd>a.exe second.c

#include <stdio.h> is a PREPROCESSOR DIRECTIVE

int is a KEYWORD

FUNCTION

main(

)

BLOCK BEGINS

int is a KEYWORD

number IDENTIFIER;

FUNCTION

printf(

"Enter an integer: " is a STRING

);

FUNCTION

scanf(

"%d" is a STRING, &

number IDENTIFIER

);

FUNCTION

if(

number IDENTIFIER %

2 is a NUMBER

== is a RELATIONAL OPERATOR

0 is a NUMBER

)

FUNCTION

printf(

"%d is even." is a STRING,

number IDENTIFIER

);

else is a KEYWORD

FUNCTION

printf(

"%d is odd." is a STRING,

number IDENTIFIER

);

return is a KEYWORD

0 is a NUMBER;

BLOCK ENDS