**G. H. RAISONI COLLEGE OF ENGG., NAGPUR**

**(An Autonomous Institute)**

**Department of Computer Science & Engg.**



**Date: 06-09-2021**

**Practical Subject: COMPILER DESIGN**

**Session: 2021-22**

**Student Details:**

| **Roll Number** | 01 |
| --- | --- |
| **Name** | Anand Suralkar |
| **Semester** | 9th |
| **Section** | A |
| **Batch** | CSE |

**Practical Details: Practical Number-1;**

| Practical Aim | Understanding of LEX specification, built in functions & variables. Write a lex program for converting a number from word to integer. |
| --- | --- |
| Theory & Syntax | LEX  Lex is a program that generates lexical analyzer. It is used with YACC parser generator.  The lexical analyzer is a program that transforms an input stream into a sequence of tokens.  It reads the input stream and produces the source code as output through implementing the lexical analyzer in the C program.  The function of Lex is as follows:  Firstly lexical analyzer creates a program lex.1 in the Lex language. Then Lex compiler runs the lex.1 program and produces a C program lex.yy.c.  Finally C compiler runs the lex.yy.c program and produces an object program a.out.  a.out is lexical analyzer that transforms an input stream into a sequence of tokens.  Specification of lex:  The lex code consists of three parts:-  • Definition section  • Rule section  • User subroutines |
| Program | %{  // take number in words as input and print its number  #include<stdio.h>  %}  %%  ZERO|zero printf("0");  ONE|one printf("1");  TWO|two printf("2");  THREE|three printf("3");  FOUR|four printf("4");  FIVE|five printf("5");  SIX|six printf("6");  SEVEN|seven printf("7");  EIGHT|eight printf("8");  NINE|nine printf("9");  %%  int main()  {  yylex();  return 0;  } |
| Output |  |
| Conclusion | Performed LEX specification, built in functions & variables And Executed a lex program for converting a number from word to integer. |