Compiler Design 19CSE401

Experiment -1A

Name: Anush E

Roll No: CH.EN.U4CSE22005

Aim:

To find the number of vowels and consonants present in the string typed by the user.

Algorithm:

1. Open gedit text editor from accessories in applications.

2. Specify the header files to be included inside the declaration part (i.e. between %{

and %}).

3. Define the digits i.e. 0-9 and identifiers a-z and A-Z.

4. Using translation rule, we defined the regular expression for digit, keywords,

identifier, operator and header file etc. if it is matched with the given input then store

and display it in yytext.

5. Inside procedure main(), use yyin() to point the current file being passed by the lexer.

6. Those specification of a lexical analyzer is prepared by creating a program lexp.l in

the LEX language.

7. The Lexp.l program is run through the LEX compiler to produce an equivalent code

in C language named Lex.yy.c .

8. The program lex.yy.c consists of a table constructed from the Regular Expressions of

Lexp.l, together with standard routines that uses the table to recognize lexemes.

9. Finally, lex.yy.c program is run through the C Compiler to produce an object program

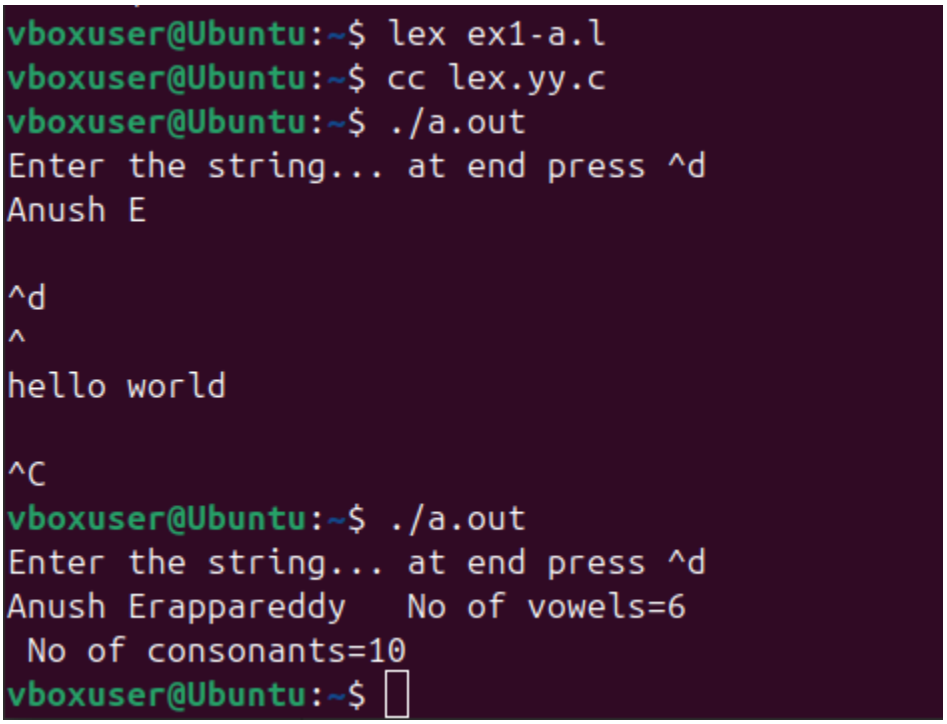
a.out, which is the lexical analyzer that transforms an input stream into a sequence of tokens.

Program Code:

```
%{

#include

int vowels=0;

int cons=0;

%}

%%

[aeiouAEIOU] {vowels++;}

[a-zA-Z] {cons++;}

%%

int yywrap() {

return 1;
```

```
}

main() {

printf(“Enter the string.. at end press ^d\n");

yylex();

printf(“No of vowels=%d\nNo of consonants=%d\n",vowels,cons);

}
```

Output:



Result:

The code has been executed and output displayed successfully