DAY-1 COMPILER DESIGN OUTPUTS

N.DATHAMADHAVA REDDY(192011450)

1. Write a LEX program to identify the capital words from the given input.

Code:

```
%%
[A-Z]+[\t\n ] { printf("%s is a capital word\n",yytext); }
. ;
%%
int main( )
{
printf("Enter String :\n");
yylex();
}
int yywrap( )
{
return 1;
}
```

2) **Write a LEX program to check whether the given input is digit or not.**

CODE:

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

%%
[0-9]    { printf("Input is a digit.\n"); }
.       { printf("Input is not a digit.\n"); }

%%

int yywrap(void) {
```

```
    return 1;
}

int main(void) {
    printf("Enter an input: ");
    yylex();
    return 0;
}
```

3) **The Company ABC runs with employees with several departments. The Organization manager had all the mobile numbers of employees. Assume that you are the manager and need to verify the valid mobile numbers because there may be some invalid numbers present. Implement a LEX program to check whether the mobile number is valid or not.**

CODE:

```
%%
[1-9][0-9]{9} {printf("\nMobile Number Valid\n");}
.+ {printf("\nMobile Number Invalid\n");}
%%
int main()
{
printf("\nEnter Mobile Number : ");
yylex();
printf("\n");
return 0;
}
int yywrap()
{ }
```

4) **. In a class, an English teacher was teaching the vowels and consonants to the students. She says "Vowel sounds allow the air to flow freely, causing the chin to drop noticeably, whilst consonant sounds are produced by restricting the air flow". As a class activity the students are asked to identify the vowels and consonants in the given word/sentence and count the number of elements in each. Write an algorithm to help the student to count the number of vowels and consonants in the given sentence.**

CODE:

```
%{
        #include<stdio.h>
        int vow=0, con=0;
%}
%%
[ \t\n]+    ;
[aeiouAEIOU]+    {vow++;}
[^aeiouAEIOU]     {con++;}
int main( )
{
        printf("Enter some input string:\n");
        yylex();
        printf("Number of vowels=%d\n",vow);
        printf("Number of consonants=%d\n",con);
}
int yywrap( )
{
return 1;
}
```

5) **Keywords are predefined, reserved words used in programming that have special meanings to the compiler. Keywords are part of the syntax and they cannot be used as an identifier. In general there are 32 keywords. The prime function of Lexical Analyser is token Generation. Among the 6 types of tokens, differentiating Keyword and Identifier is a challenging issue. Thus write a LEX program to separate keywords and identifiers.**

Code:

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

/* List of C language keywords, expand or modify according to needs */
KEYWORDS    "auto"|"break"|"case"|"char"|"const"|"continue"|"default"|"do"|
        "double"|"else"|"enum"|"extern"|"float"|"for"|"goto"|"if"|
        "int"|"long"|"register"|"return"|"short"|"signed"|"sizeof"|
        "static"|"struct"|"switch"|"typedef"|"union"|"unsigned"|
        "void"|"volatile"|"while"

/* Identifier rule (simplified version, adjust as needed) */
IDENTIFIER  [a-zA-Z_][a-zA-Z0-9_]*

%%

{KEYWORDS}  { printf("Keyword: %s\n", yytext); }
{IDENTIFIER} { printf("Identifier: %s\n", yytext); }

[ \t\n]+    ; /* Ignore whitespaces */

.       { printf("Unknown: %s\n", yytext); } /* Catch all for other characters */

%%

int yywrap() { return 1; }
```

```
int main(int argc, char **argv) {
    if (argc > 1) {
        if (!freopen(argv[1], "r", stdin)) {
            printf("Could not open file %s\n", argv[1]);
            return 1;
        }
    }

    yylex();

    return 0;
}
```

6) **Write a LEX program to identify and count positive and negative numbers.**

CODE:

```
%{
#include <stdio.h>
int positive_count = 0, negative_count = 0;
%}
%%
[ \t\n]+           ; /* Ignore whitespace */
-?[1-9][0-9]*      { /* Match positive and negative integers */
   if (yytext[0] == '-') {
      negative_count++;
   } else {
      positive_count++;
   }
}
%%
int main() {
   printf("Enter some numbers (end with EOF):\n");
   yylex();
   printf("Positive numbers: %d\n", positive_count);
   printf("Negative numbers: %d\n", negative_count);
   return 0;
}

int yywrap() {
   return 1;
}
```

7)Write a LEX program to recognise numbers and words in a statement. Pooja is a small girl of age 3 always fond of games. Due to the pandemic, she was not allowed to play outside. So her mother designs a gaming event by showing a flash card. Pooja has to separate the numbers in one list and words in another list shown in the flash card.

CODE:

```
%{
#include <stdio.h>
#include <string.h>
%}
WORD    [a-zA-Z]+
NUMBER  [0-9]+
%%
{WORD}  {
          printf("Word: %s\n", yytext);
       }
{NUMBER} {
          printf("Number: %s\n", yytext);
       }
\n|.    ;
%%
int yywrap() {
   return 1;
}
int main() {
   printf("Enter a statement for Pooja:\n");
   yylex();
   return 0;
}
```

8) Write a LEX program to accept string starting with vowel.

CODE:

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

%%
^[aeiouAEIOU].* {
   printf("Accepted: %s\n", yytext);
}
.|\n {
   printf("Rejected: %s\n", yytext);
}

%%

int yywrap(void) {
   return 1;
}

int main(void) {
   printf("Enter a string: ");
   yylex();
   return 0;
}
```