CD lab 5

name: Abhinav Singh Bhagtana
reg no: 230905225
roll no: A-029

Q1.Count the number of vowels and consonants in the given input.

```
%{
#include <stdio.h>
int vowels = 0;
int consonants = 0;
%}

%%

[aeiouAEIOU]         { vowels++; }
[a-zA-Z]             { consonants++; }
.                    { }

%%

int main() {
    printf("Enter a string: \n");
    yylex();
    printf("\n--- Results ---\n");
    printf("Vowels: %d\n", vowels);
    printf("Consonants: %d\n", consonants);
    return 0;
}

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

output:

```
Enter a string:
hi hello my name


--- Results ---
Vowels: 5
Consonants: 8
```

Q2.Count the number of words, characters, blanks and lines in a given text.

```
%{
#include <stdio.h>
int words = 0;
int chars = 0;
int blanks = 0;
```

```
int lines = 0;
%}

%%
[a-zA-Z_][a-zA-Z0-9_]* { words++; chars += yyleng; }
[ ]+                   { blanks += yyleng; }
\n                     { lines++; }
.                      { chars++; }
%%

int main() {
    printf("Enter string (Ctrl+D to finish):\n");
    yylex();
    printf("\n--- Results ---\n");
    printf("Words: %d\nChars: %d\nBlanks: %d\nLines: %d\n", words, chars, blanks, lines);
    return 0;
}

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

output:

```
Enter string (Ctrl+D to finish):
hi hello
my name

--- Results ---
Words: 4
Chars: 13
Blanks: 2
Lines: 2
```

Q3.Find the number of positive integer, negative integer, positive floating positive number and negative floating point number

```
%{
#include <stdio.h>
int posint = 0;
int negint = 0;
int floatpos = 0;
int floatneg = 0;
%}

%%

-[0-9]+\.[0-9]+    { floatneg++; }


\+?[0-9]+\.[0-9]+  { floatpos++; }
```

```
-[0-9]+          { negint++; }


\+?[0-9]+        { posint++; }

[ \t\n]          {  }
.               { }
%%

int main() {
   printf("Enter numbers (Ctrl+D to finish):\n");
   yylex();
   printf("\n--- Summary ---\n");
   printf("Positive integers: %d\n", posint);
   printf("Negative integers: %d\n", negint);
   printf("Positive floating point: %d\n", floatpos);
   printf("Negative floating point: %d\n", floatneg);
   return 0;
}

int yywrap() { return 1; }
```
output:
```
Enter numbers (Ctrl+D to finish):
54
34
24.4
-23
-2333
-0.09

--- Summary ---
Positive integers: 2
Negative integers: 2
Positive floating point: 1
Negative floating point: 1
```

Q4.Given a input C file, replace all scanf with READ and printf with WRITE statements also find the number
of scanf and printf in the file.

```
%{
#include <stdio.h>
int s_count = 0;
int p_count = 0;
%}

%%

"scanf"     { s_count++; fprintf(yyout, "READ"); }
```

```
"printf"    { p_count++; fprintf(yyout, "WRITE"); }

%%

int main(int argc, char **argv) {
    if (argc < 3) {
        printf("Usage: %s <input_file> <output_file>\n", argv[0]);
        return 1;
    }


    yyin = fopen(argv[1], "r");
    if (!yyin) {
        perror("Error opening input file");
        return 1;
    }


    yyout = fopen(argv[2], "w");
    if (!yyout) {
        perror("Error creating output file");
        return 1;
    }

    yylex();


    fprintf(yyout, "\n\n/* --- Statistics --- */\n");
    fprintf(yyout, "/* Total scanf replaced: %d */\n", s_count);
    fprintf(yyout, "/* Total printf replaced: %d */\n", p_count);

    fclose(yyin);
    fclose(yyout);

    printf("Processing complete. Check %s for results.\n", argv[2]);
    return 0;
}

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

input:

```
#include <stdio.h>
int main()
{
        int n;
        printf("hello world");
        scanf("%d",&n);
```

```
        printf("taken input");
        printf("output %d",n);
        return 0;
}
```
output:
```
#include <stdio.h>
int main()
{
        int n;
        WRITE("hello world");
        READ("%d",&n);
        WRITE("taken input");
        WRITE("output %d",n);
        return 0;
}


/* --- Statistics --- */
/* Total scanf replaced: 1 */
/* Total printf replaced: 3 */
```

Q5.That changes a number from decimal to hexadecimal notation.

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

int num[100];
int i = 0;
%}

%%
[0-9]+  {

         num[i++] = atoi(yytext);
       }
[ \t\n] {  }
.       { }
%%

void printHex(int n) {
   if (n == 0) {
      printf("0");
      return;
   }

   char hex[50];
   int j = 0;
```

```
    while (n > 0) {
        int temp = n % 16;
        if (temp < 10)
            hex[j++] = temp + 48; // '0'-'9'
        else
            hex[j++] = temp + 55; // 'A'-'F'
        n = n / 16;
    }


    for (int k = j - 1; k >= 0; k--) {
        printf("%c", hex[k]);
    }
}

int main() {
    printf("Enter numbers (Ctrl+D to finish):\n");
    yylex();

    printf("\n--- Hexadecimal Results ---\n");
    for(int k = 0; k < i; k++) {
        printf("Decimal %d to Hex: ", num[k]);
        printHex(num[k]);
        printf("\n");
    }
    return 0;
}

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

output:

```
Enter numbers (Ctrl+D to finish):
234
234
233
33
2

--- Hexadecimal Results ---
Decimal 234 to Hex: EA
Decimal 234 to Hex: EA
Decimal 233 to Hex: E9
Decimal 33 to Hex: 21
Decimal 2 to Hex: 2
```

Q6.Convert uppercase characters to lowercase characters of C file excluding the characters present in the
comment.

```
%{
#include <stdio.h>
#include <ctype.h>
int count = 0;
%}


%x COMMENT

%%


"//".* { fprintf(yyout, "%s", yytext); }


"/*"              { fprintf(yyout, "%s", yytext); BEGIN(COMMENT); }
<COMMENT>"*/"      { fprintf(yyout, "%s", yytext); BEGIN(INITIAL); }
<COMMENT>.         { fputc(yytext[0], yyout); }
<COMMENT>\n        { fputc('\n', yyout); }


[A-Z]            {
                  count++;
                  fputc(tolower(yytext[0]), yyout);
                 }


.                { fputc(yytext[0], yyout); }
\n               { fputc('\n', yyout); }

%%

int main(int argc, char **argv) {
    if (argc < 3) {
        printf("Usage: %s <input_file> <output_file>\n", argv[0]);
        return 1;
    }

    yyin = fopen(argv[1], "r");
    yyout = fopen(argv[2], "w");

    if (!yyin || !yyout) {
        perror("File error");
        return 1;
    }

    yylex();
```

```
    fprintf(yyout, "\n\n/* --- Statistics --- */\n");
    fprintf(yyout, "/* Total uppercase characters replaced: %d */\n", count);

    fclose(yyin);
    fclose(yyout);
    return 0;
}

int yywrap() { return 1; }
```
input:
```
#INCLUDE <stdio.h>
//INCLUDED STDIO
int MAIN(){
        RETURN 1;
}
```
output:
```
#include <stdio.h>
//INCLUDED STDIO
int main(){
        return 1;
}


/* --- Statistics --- */
/* Total uppercase characters replaced: 17 */
```