AIM: Write a lex program to count positive and negative numbers from the input file. (Note: It is compulsory to read the input from the file and display the results in another file)

CODE:

Input.txt

```
6 -10 9.6 -17.1 1-e -2-e
```

Practical_3.l

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

int pos_int_count = 0;
int neg_int_count = 0;
int pos_float_count = 0;
int neg_float_count = 0;
int pos_exp_int_count = 0;
int neg_exp_int_count = 0;
int pos_exp_float_count = 0;
int neg_exp_float_count = 0;
%}

%%
[+]?[0-9]+[eE][-]?[0-9]*           { pos_exp_int_count++; }
[+]?[0-9]+"-e"                     { pos_exp_int_count++; }
-[0-9]+[eE][-]?[0-9]*              { neg_exp_int_count++; }
-[0-9]+"-e"                        { neg_exp_int_count++; }
[+]?[0-9]*\.[0-9]+[eE][-]?[0-9]*   { pos_exp_float_count++; }
[+]?[0-9]*\.[0-9]+"-e"             { pos_exp_float_count++; }
-[0-9]*\.[0-9]+[eE][-]?[0-9]*      { neg_exp_float_count++; }
-[0-9]*\.[0-9]+"-e"                { neg_exp_float_count++; }
[+]?[0-9]+                         { pos_int_count++; }
-[0-9]+                            { neg_int_count++; }
[+]?[0-9]*\.[0-9]+                 { pos_float_count++; }
-[0-9]*\.[0-9]+                    { neg_float_count++; }
[ \t\n]                            ; // Ignore whitespace
.                                  ; // Ignore any other character
%%

int main(int argc, char **argv) {
    if (argc != 3) {
        fprintf(stderr, "Usage: %s <input file> <output file>\n", argv[0]);
        return 1;
    }

    FILE *infile = fopen(argv[1], "r");
    FILE *outfile = fopen(argv[2], "w");
```

```
    if (!infile || !outfile) {
        perror("File error");
        return 1;
    }



    yyin = infile;
    yyout = outfile;

    yylex();

    fprintf(outfile, "Positive integers: %d\n", pos_int_count);
    fprintf(outfile, "Negative integers: %d\n", neg_int_count);
    fprintf(outfile, "Positive floats: %d\n", pos_float_count);
    fprintf(outfile, "Negative floats: %d\n", neg_float_count);
    fprintf(outfile, "Positive exponential integers: %d\n", pos_exp_int_count);
    fprintf(outfile, "Negative exponential integers: %d\n", neg_exp_int_count);
    fprintf(outfile, "Positive exponential floats: %d\n", pos_exp_float_count);
    fprintf(outfile, "Negative exponential floats: %d\n", neg_exp_float_count);

    fclose(infile);
    fclose(outfile);

    return 0;
}
```

OUTPUT:

```
[21012021001@linuxserv ~]$cat input.txt
6 -10 9.6 -17.1 1-e -2-e
```

```
[21012021001@linuxserv ~]$ lex cd_p3.l
[21012021001@linuxserv ~]$ gcc lex.yy.c -o count_numbers -ll
[21012021001@linuxserv ~]$ ./count_numbers input.txt output.txt
[21012021001@linuxserv ~]$ cat output.txt
Positive integers: 1
Negative integers: 1
Positive floats: 1
Negative floats: 1
Positive exponential integers: 1
Negative exponential integers: 1
Positive exponential floats: 0
Negative exponential floats: 0
```