PROGRAM TITLE: Read a file containing text and print its contents in reverse order (bottom-up fashion). Give the file name as a command line argument.

PROGRAM ALGORITHM:

```
algo fileop(parameters: argc,argv)
{
    /*argc and argv are both command line arguments. Argv is a string
array while argc is the number of arguments*/
    check if filename entered or not and display corresponding message
    load the file pointer
    while(file pointer doesnt reach end of file)
    {
        print the contents of the file
    }
    set count equal to number of characters in file
    while(count not equal to -1)
    {
        traverse file in the opposite direction
        decrease count
    }
    close the original file
}
```

PROGRAM CODE:

```c
/*C Program to print the reverse of a File */
#include <stdio.h>
#include <stdlib.h>
int main(int argc,char *argv[])
{
    FILE *fp;
    int i;
    char ch;
    long count;

    /*Check if file name entered or not*/
    if(argc!=2)
    {
        printf("You forgot to enter the filename\n");
        exit(1);
    }

    /*Display error message if file pointer could not be loaded*/
    if((fp=fopen(argv[1],"r"))==NULL)
    {
        printf("Cannot open file\n");
        exit(2);
    }

    /*Display the contents of the file*/
    printf("The file is:\n");
```

```c
    ch=getc(fp);
    while(ch!=EOF)
    {
        putchar(ch);
        ch=getc(fp);
    }

    /*Take file pointer to EOF*/
    fseek(fp,-1L,SEEK_END);
    count=ftell(fp);
    printf("The Reverse of the file is:");

    /*Print reverse of the file*/
    while(count!=-1)
    {
        ch=getc(fp);
        putchar(ch);
        fseek(fp,-2L,SEEK_CUR);
        count--;
    }
    printf("\n");

    /*Close the original file to avoid damage*/
    fclose(fp);
    return 0;
}
```

OUTPUT:

DISCUSSION:

The command lines can be given as **./a.out "text file"** or **./a.out text\ file** . The file name has to be provided in the command line as an argument.