**Linux shell:**

cat `find . -name \\*.sh` |tr -d " \t"|grep .|grep -v ^#|wc -1

find . -name \\*.sh: find files matching the pattern, list with path

cat '': list contents of these files

tr -d " \t": trim tabs and spaces

grep .: remove empty lines

grep -v ^#: remove comments

wc -1: count lines of output

**C code:**

/\* C Program to count the Number of Lines in a C File \*/

#include <stdio.h>

#define MAX\_FILE\_NAME 100

int main()

{

FILE \*fp;

int count = 0,count1=0,count2;

// Line counter (result)

char filename[MAX\_FILE\_NAME];

char c;

// To store a character read from file

// Get file name from user. The file should be

// either in current folder or complete path should be provided

printf("Enter file name: ");

scanf("%s", filename);

// Open the file

fp = fopen(filename, "r");

// Check if file exists

if (fp == NULL)

{

printf("Could not open file %s", filename);

return 0;

}

// Extract characters from file and store in character c

for (c = getc(fp); c != EOF; c = getc(fp))

{ if (c == '\n') // Increment count if this character is newline

count = count + 1;

if (c == '/')

{

if ((c = fgetc(fp)) == '/')

{ count1=count1+1;

}

}

}

// Close the file

fclose(fp);

count2=count-count1;

printf("The file %s has total of %d lines\n ", filename, count);

printf("The file %s has %d comment lines\n ", filename, count1);

printf("The file %s has total of %d lines excluding comment lines\n ", filename, count2);

return 0;

}