

\*\*\*\*\*\*\*\*\*\*\*\*\*prog.h\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

typedef struct {

int total\_chars;

int letters\_count;

int words\_count;

int lines\_count;

int max\_line\_length;

} FileStats;

int is\_letter(char);

FileStats\* process\_file(char\*);

char\*\* get\_lines(char\*);

void write\_rev(char\*, char\*\*, int);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*prog.c\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <stdio.h>

#include <stdlib.h>

#include "prog.h"

int main(){

FileStats \*fsp;

char\*\* lines;

fsp = process\_file("wcs.txt");

if (fsp){

printf("File size = %d\n", fsp->total\_chars);

printf("Number of letters = %d\n", fsp->letters\_count);

printf("Number of Words = %d\n", fsp->words\_count);

printf("Number of lines = %d\n", fsp->lines\_count);

printf("Length of longest line = %d\n", fsp->max\_line\_length);

}

lines = get\_lines("wcs.txt");

write\_rev("wcs-rev.txt", lines, fsp->lines\_count);

return 0;

}

int is\_letter(char c){

return (c>='A' && c<='Z') || (c>='a' && c<='z');

}

FileStats\* process\_file(char\* fn){

FileStats\* fs;

FILE\* fp;

char c;

int inword = 0;

int line\_length = 0;

if (!(fs = (FileStats\*)calloc(1, sizeof(FileStats)))) return NULL;

if (!(fp = fopen(fn, "r"))) return NULL;

do {

c=getc(fp);

fs->total\_chars++;

if (is\_letter(c)){

fs->letters\_count++;

inword = 1;

}

else {

if (inword) fs->words\_count++;

inword = 0;

}

if (c == '\n' || c == EOF){

fs->lines\_count++;

if (fs->max\_line\_length < line\_length)

fs->max\_line\_length = line\_length;

line\_length = 0;

}

else line\_length++;

} while (c != EOF);

fclose(fp);

return fs;

}

char\*\* get\_lines(char\* fn){

char c;

int i;

char \*\*lines, \*p;

FileStats\* fs;

FILE\* fp;

if (!(fs = process\_file(fn))) return NULL;

lines = (char\*\*)calloc(fs->lines\_count, sizeof(char\*));

for (i=0; i < fs->lines\_count; i++)

lines[i] = (char\*)calloc(fs->max\_line\_length+1, 1);

if (!(fp = fopen(fn, "r"))) return NULL;

i = 0;

p = lines[0];

do {

c = getc(fp);

if (c == '\n' || c == EOF)

p = lines[++i];

else

\*p++ = c;

} while (c != EOF);

return lines;

}

void write\_rev(char\* fn, char\*\* lines, int count){

int i;

FILE\* fp = fopen(fn, "w");

for (i=count-1; i >= 0; i--){

fputs(lines[i], fp);

if (i) fprintf(fp, "\n");

}

fclose(fp);

}