

	<b>Carátula para entrega de prácticas</b>	
Facultad de Ingeniería	Laboratorio de docencia	

# Laboratorios de computación salas A y B

<i>Profesor:</i>	M.C. Alejandro Esteban Pimentel Alarcón
<i>Asignatura:</i>	Fundamentos de programación
<i>Grupo:</i>	3
<i>No de Práctica(s):</i>	13
<i>Integrante(s):</i>	Vázquez Espinosa Ximena Itzel Aguilar Lara Alexa Patricia
<i>No. de Equipo de cómputo empleado:</i>	33
<i>No. de Lista o Brigada:</i>	8015 5515
<i>Semestre:</i>	Primer semestre
<i>Fecha de entrega:</i>	11 de noviembre 2019
<i>Observaciones:</i>	

CALIFICACIÓN: \_\_\_\_\_

# Lectura y escritura de datos

Para la resolución de ésta práctica, se tendrá que aprender a manejar los archivos por medio del código c, donde se utilizarán todos los conocimientos previos para mezclarlos con los nuevos, puesto que, conociendo la funcionalidad de los elementos previos, se nos facilitará comprender lo que conlleva esta práctica.

Objetivo: Elaborar programas en lenguaje C que requieran el uso de archivos de texto plano en la resolución de problemas, entendiendo a los archivos como un elemento de almacenamiento secundario.

## Archivos

Para trabajar con archivos en C, es necesario tener un apuntador hacia un archivo:

**FILE \*archivo;**

Para asignar el apuntador a su lugar correspondiente, podemos contar con una función para abrir el archivo por nombre:

**archivo=fopen("archivo.txt", "r");**

A partir de este punto, ya podemos utilizar nuestro apuntador de archivo. Pero para leer, necesitaremos un variable en dónde guardar el texto:

**char linea[90];**

**char palabra[9];**

## Formas para abrir

Al momento de abrir un archivo (*fopen*) se puede elegir una entre varias opciones:

r: Abre un archivo de texto para lectura.

w: Crea un archivo de texto para escritura.

a: Abre un archivo de texto para añadir.

r+: Abre un archivo de texto para lectura / escritura.

w+: Crea un archivo de texto para lectura / escritura.

a+: Añade o crea un archivo de texto para lectura / escritura.

## string.h

String en una librería que será de utilidad siempre que tengamos que manejar un texto (cadenas de caracteres).

**#include <string.h>**

**strlen(char texto[]);**

**strcpy(char destino[], char origen []);**

**strcmp(char texto [], char texto 2[]);**

**strcat(char destino[], char origen []);**

**strstr(char texto[], char buscando []);**

**strchr(char texto[], char buscando []);**

El secreto de las cadenas de caracteres es que indican su final con el caracter nulo "\0". Pero deben recordar que ese caracter también ocupa un lugar en el arreglo.

## Leer

Tenemos dos formas sencillas de leer texto desde un archivo:

**fscanf(archivo, "%8s", palabra);**

Que funciona igual que *scanf()* con la diferencia de que como primer parámetro recibe el apuntador hacia el archivo. Recuerden que esto solo lee una palabra a la vez.

Y la otra manera es:

**fgets(linea, 89, archivo);**

Que lee una línea completa, con un número máximo de caracteres que recibe como segundo argumento. Este número es útil para no sobrepasar la longitud del arreglo de caracteres.

Observen que el orden del apuntador y la variable se invierten.

Ambas formas harán que el archivo “avance”. Eso quiere decir que si las usan repetidas veces, irán leyendo nuevas palabras/líneas del archivo.

```
while( ! feof(archivo) ){  
    fgets(linea, 80, archivo);  
    printf("%s", linea);  
}
```

La función *feof* nos ayuda a darnos cuenta si ya llegamos al final del archivo. La función recibe como único argumento el apuntador al archivo, y devuelve 0 mientras no sea el final del archivo.

## Escribir

También hay varias formas de escribir en un archivo, pero por familiaridad, la más sencilla es *fprintf*.

**fprintf(archivo, "%sn%", "texto");**

Al igual que con *fscanf*, lo único que cambia es que el primer parámetro es el apuntador del archivo.

## Cerrar

Por último, hay que cerrar el archivo que abrimos con *fopen*. Se cierra con *fclose*.

**fclose(archivo);**

## Actividad

Crear un programa que pida el nombre de un archivo de entrada y un archivo de salida.

Para el archivo de entrada, mostrar:

- Texto.
- Número de líneas.
- Número de palabras (cualquier cosa entre espacios)
- Número de caracteres.

Para el archivo de salida:

- Copiar el archivo de entrada con las líneas invertidas.

```

#include <stdio.h>
#include <string.h>

int main (){
    FILE * archivo, *archivosalida;
    archivo = fopen("calaverita.txt", "r");

    char linea[100];
    char palabra[20];
    int lineas;

    int contador = 0;

    int longui=0;
    int caracteres;

    lineas=0;

    archivo = fopen("calaverita.txt", "r");
    while(!feof(archivo)){
        fgets(linea, 99, archivo);
        printf("%s", linea);
        lineas=lineas+1;
    }
    printf("Número de líneas: %i\n", lineas);

    archivo = fopen("calaverita.txt", "r");
    while(!feof(archivo)){
        fscanf(archivo, "%8s", palabra);
        contador++;
    }

    printf("Número de palabras: %i \n", contador);

    archivo = fopen("calaverita.txt", "r");
    while(!feof(archivo)){
        fgets(linea, 89, archivo);
        longui=strlen(palabra);
        caracteres=caracteres + longui;
    }
    printf("Número de caracteres %i \n", caracteres);

    archivo=fopen("calaverita.txt", "r");
    archivosalida=fopen("calaverita_salida.txt", "w");
    char listaarchivo[lineas][100];

    for(int i=0; i<lineas; i++){
        if(i==0){
            printf(archivosalida, "%s \n", listaarchivo[i]);
        }
        else{

```

```

        fprintf(archivosalida, "%s", listaarchivo[i]);
    }
}
return 0;
}

```

```

1  #include <stdio.h>
2  #include <string.h>
3
4  int main (){
5      FILE * archivo, *archivosalida;
6      archivo = fopen("calaverita.txt", "r");
7
8      char linea[100];
9      char palabra[20];
10     int lineas;
11
12     int contador = 0;
13
14     int longui=0;
15     int caracteres;
16
17     lineas=0;
18
19     archivo = fopen("calaverita.txt", "r");
20     while(!feof(archivo)){
21         fgets(linea, 99, archivo);
22         printf("%s", linea);
23         lineas=lineas+1;
24     }
25     printf("Número de líneas: %i\n", lineas);
26
27     archivo = fopen("calaverita.txt", "r");
28     while(!feof(archivo)){
29         fscanf(archivo, "%8s", palabra);
30         contador++;
31     }
32
33     printf("Número de palabras: %i \n", contador);
34
35     archivo = fopen("calaverita.txt", "r");
36     while(!feof(archivo)){
37         fgets(linea, 89, archivo);
38         longui=strlen(palabra);
39         caracteres=caracteres + longui;
40     }
41     printf("Número de caracteres %i \n", caracteres);
42
43     archivo=fopen("calaverita.txt", "r");
44     archivosalida=fopen("calaverita_salida.txt", "w");
45     char listaarchivo[lineas][100];
46
47     for(int i=0; i<lineas; i++){
48         if(i==0){
49             fprintf(archivosalida, "%s \n", listaarchivo[i]);
50         }
51         else{
52             fprintf(archivosalida, "%s", listaarchivo[i]);
53         }
54     }
55     return 0;
56 }

```



```
Descargas — -bash — 80x60
[Paraguay33:downloads fp03alu52$ ./main
En este mes singular,
En que todo es fiesta y danza,
Recordamos con amor
Tradiciones y alabanzas.

Es menester recordar
Que aún en la confianza,
De volvernos a encontrar,
Recordamos sus andanzas.

A nuestros seres queridos
Les ponemos un altar,
Pues sus almas y latidos
Los sentimos regresar!

Así que los festejamos
Con grandes piezas de pan,
Colocados en altares
Con flores velas, mezcal!

Estos panes primorosos,
Que de Colores están,
Con formas de cuerno y hueso
Se los hemos de dejar.

Para que pasen contentos
En su visita fugaz,
En esta tierra de amores
Los recuerdos que se van.

Para mí es un gran regalo
El poderlos encontrar,
En fechas tan especiales
Y poderlos disfrutar.

Si supieran mis muertitos
Cuánto los echo de menos,
Vendrían todos los días
Y se harían muy amenos!

Yo los espero sentada
Pues sé que departiremos
Estas dos noches completas
Con atoles y galletas.

Queridas almas contentas,
De saber que las queremos,
Recordamos sus amores,
Y esperamos su regreso!

Autora: Davina Gpe. Ponce Mtz.

Número de líneas: 55
Número de palabras: 198
Número de caracteres 220
Paraguay33:downloads fp03alu52$ █
```

En conclusión, es importante comprender los fundamentos de programación, así como la lógica que llevará a cabo la computadora para poder comprender cómo manejar archivos en C, puesto que los recursos utilizados son muy parecidos se facilita la comprensión, la cual de otra manera se volvería de compleja para nuestro entendimiento.