

**Practica 12**

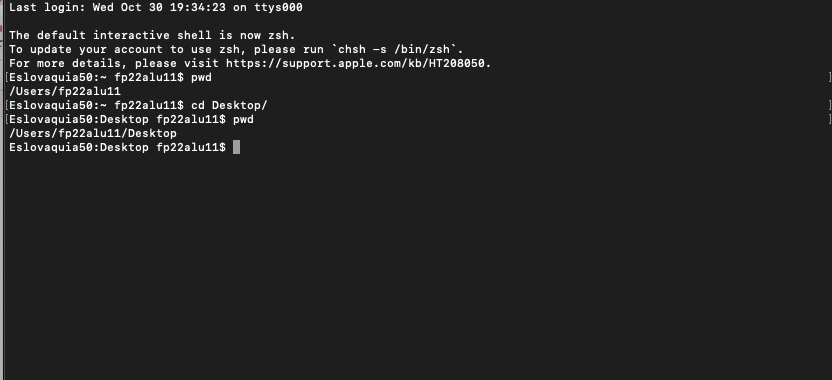
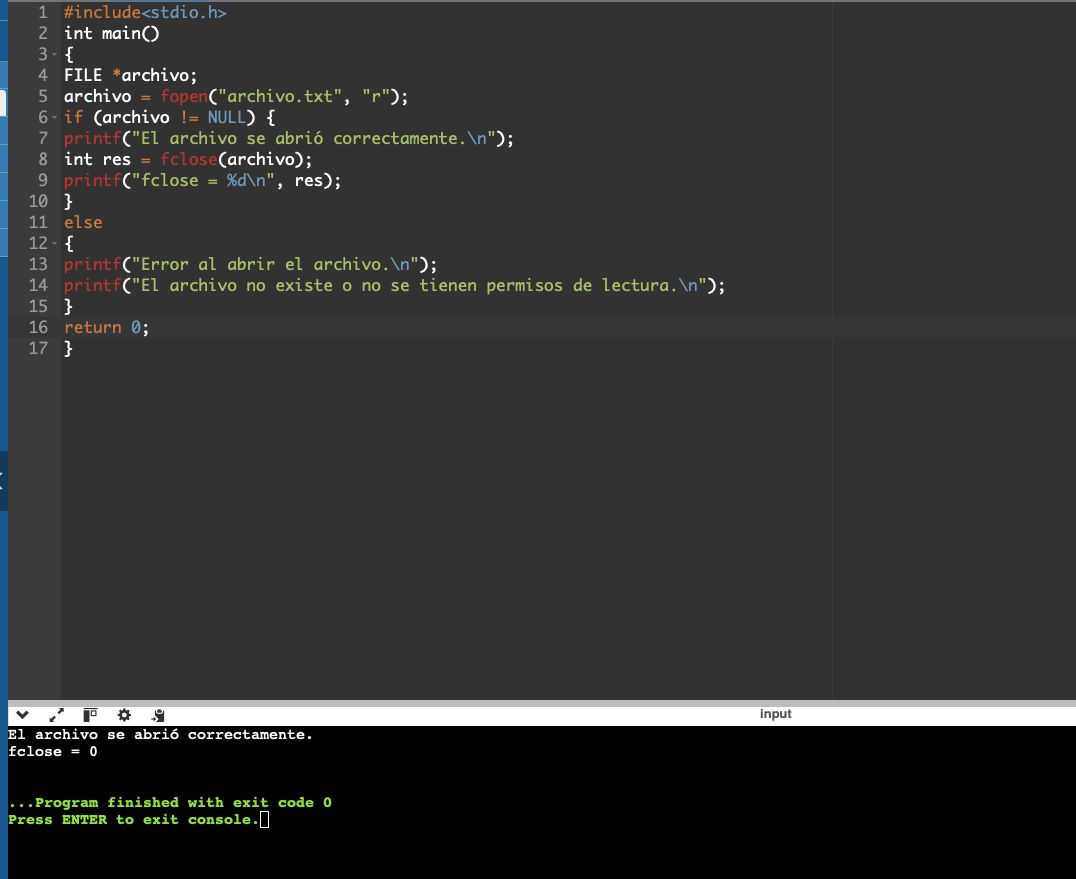
Lectura y escritura de datos

**Objetivo:**

El alumno elaborará 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.

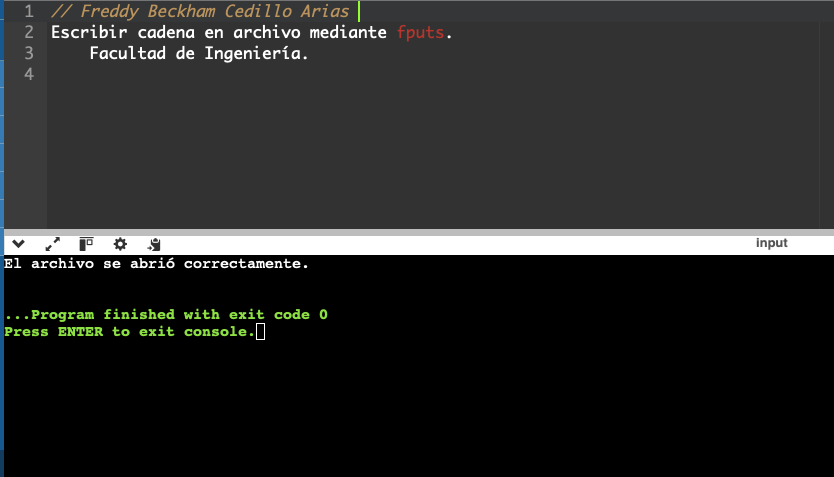
**Desarrollo:**

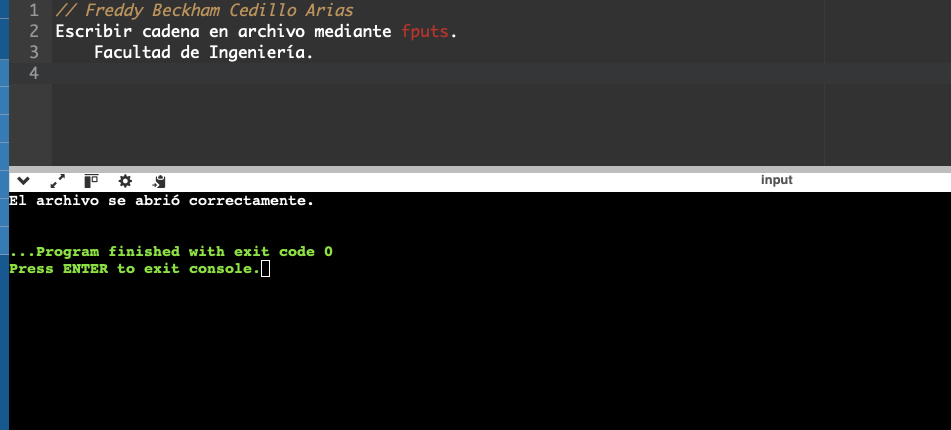
Programa1.c

Se hizo el cambio de “r” por “w” para poder crear el archivo, además de utilizar fopen y fclose para crear el archivo y cerrar el archivo creado respectivamente. 

Programa 3.c

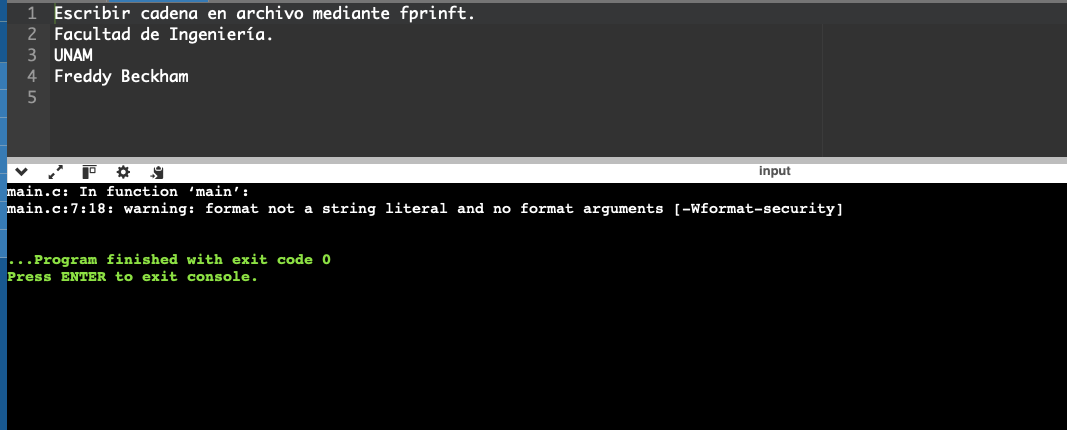
Se utilizaron las funciones fopen y fclose para abrir y cerrar el archivo. Además de que se utilizo fputs para poder escribir en el archivo.

Programa 2.c

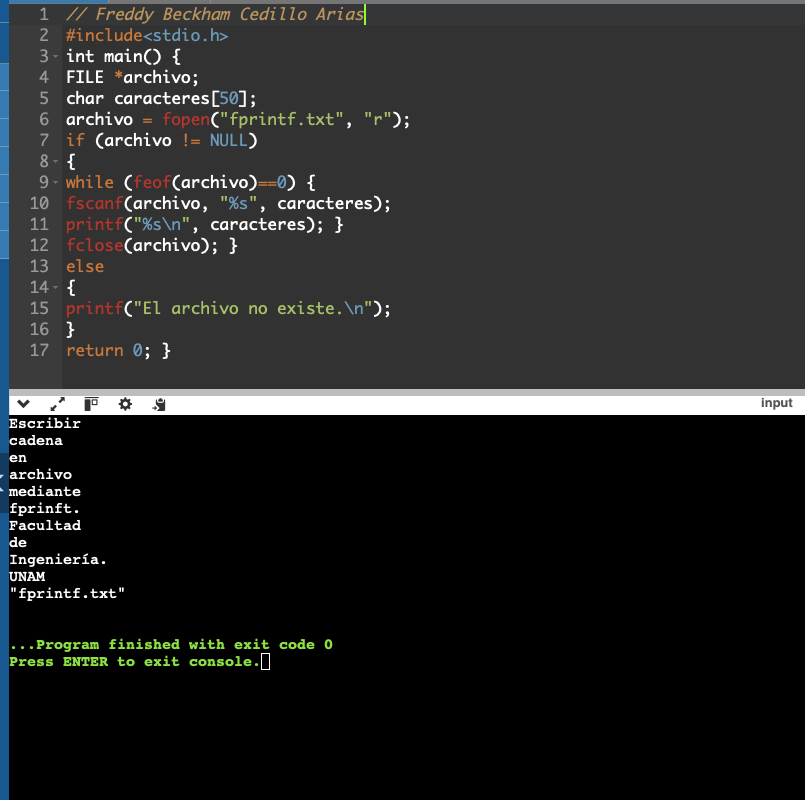
Se emplearon fopen y fclose para crear un archivo y cerrarlo respectivamente y fgets para leerlo. 

Programa 5.c

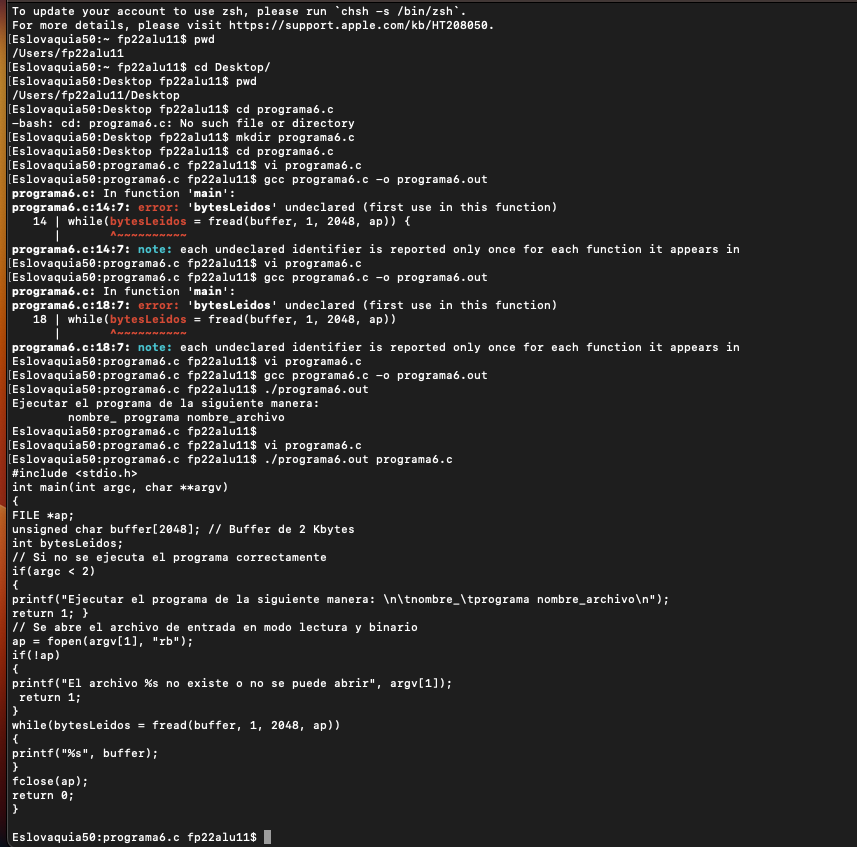
Se utilizaron fopen y fclose para abrir y cerrar el archivo, para después usar fprintf para imprimir en el archivo.

Programa 4.c

Se emplearon fopen y fclose para abrir y cerrar el archivo, y posteriormente fscanf para leerlo.



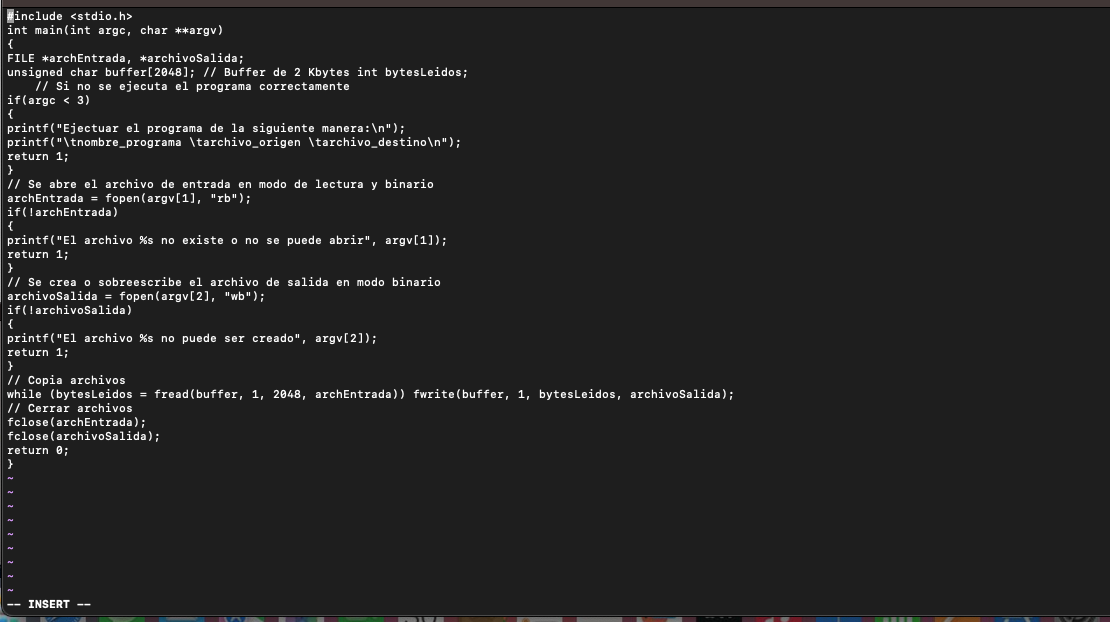
Se ejecutaron los programas 6.c y 7.c en la terminal

****

Sintaxis de [funciones](https://tuaulavirtual.educatic.unam.mx/mod/resource/view.php?id=1184966).:

int fnNombredelprograma()

{

//Bloque de código de la función

}

[Funciones](https://tuaulavirtual.educatic.unam.mx/mod/resource/view.php?id=1184966) de lectura y escritura en un archivo:

La función fgets() permite leer un un archivo y la función fputs() permite escribir en un archivo y la sintaxis de ambas aparece en la tabla

**Tarea:**

1.- Completa el cuadro referente a [funciones](https://tuaulavirtual.educatic.unam.mx/mod/resource/view.php?id=1184966), sintaxis, ejemplo y características de cada una de las [funciones](https://tuaulavirtual.educatic.unam.mx/mod/resource/view.php?id=1184966)vistas en el laboratorio.

| Función | Sintaxis | Características | Ejemplo de sintaxis |
| --- | --- | --- | --- |
| fputs() | fputs char \*fputs(char  \*buffer, FILE  \*apArch); | La función fputs()  permite escribir una  cadena en un archivo  especifico | fputs (listaNombre,  archivo); |
| fgets() | char\*fputs(char\*buffer,FILE  \*apArch) | Permite leer una cadena  de código del archivo  especificado y solo lee un renglón a la vez. | Fputs (escribir,  archivo); |
| fopen() | \*FILEfopen(char  \*nombre\_archivo,char  \*modo); | Existen diferentes modos en los que se puede abrir el archivo y se puede usar más de uno. | =fopen(“archivo.txt”  ,  “r”); |
| fclose() | int fclose(FILE\*apArch); | Finaliza el archivo que se  abrió, si se cierra de  forma incorrecta se puede dañar el archivo o incluso destruirse. | Int res =  fclose(archivo); |
| fprintf() | int fprintf(FILE\*apArch,char  \*formato, ...); | apArch es un apuntador  el cual al archivo devuelto por una función llamada fopen(), que funciona de la misma manera que printf. | Fprintf(archivo,  escribir); |
| fscanf() | int  fscanf(FILE\*apArch,char  \*formato, ...); | Del mismo modo aquí  apArch es un apuntador,  el cual sirve de la misma  manera que scanf. | Fscanf(archivo, “%s”  ,  caracteres); |
| fread() | int fread(void\*ap,size\_t  tam,size\_tnelem,FILE\*archivo) | Permite leer a partir de  un apuntador uno o varios elementos de la misma longitud. | Fread(buffer, 1, 2048,  ap) |
| fwrite() | int fwrite(void\*ap,size\_t  tam,  size\_tnelem,FILE\*archivo) | Permite escribir hacia un  archivo uno o varios  elementos de la misma  magnitud que se  encuentran almacenados en un apuntador. | Fwrite(buffer, 1,  bytesLeidos,  archivoSalida); |

**Conclucion:**

Las funciones son muy útiles porque permiten dividir un programa en partes más pequeñas y fáciles de manejar. Nos ayudan a organizar nuestro código, evitar repetirlo y resolver problemas de manera más fácil. Las funciones nos será útil en el futuro ya que esta herramienta será útil para hacer programas más claras y fáciles al desarrollar.

**Bibliografia:**

(No se hizo uso de investigación en la web)