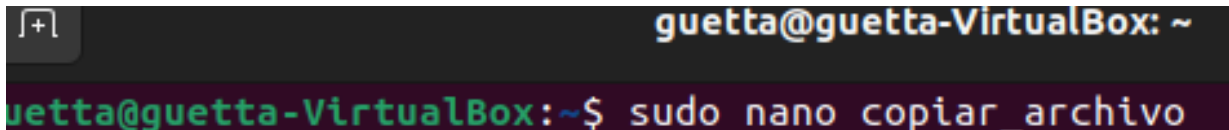


# Laboratorio 1: Introducción a la Gestión de archivos en C

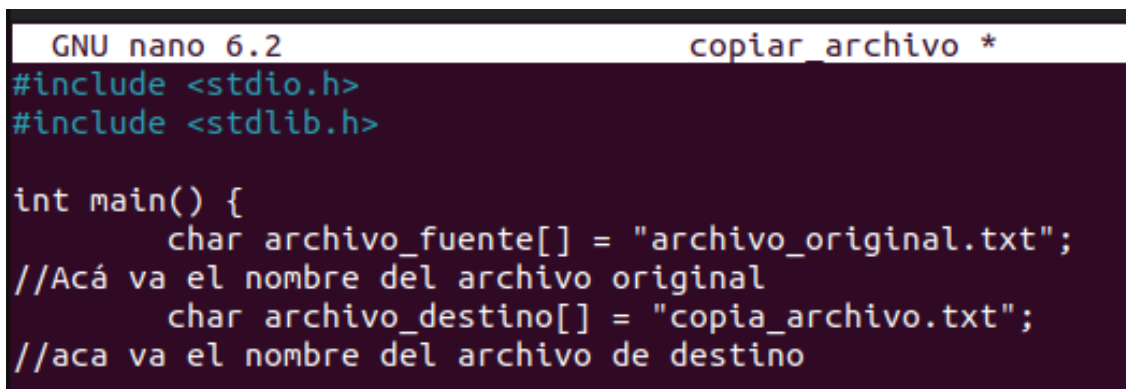
## Luis David Mosquera Donado

Primeramente, se creó un archivo con el comando “sudo nano copia\_archivo” para entrar a donde voy a crear el código.



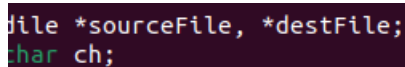
```
guetta@guetta-VirtualBox: ~  
guetta@guetta-VirtualBox:~$ sudo nano copia_archivo
```

Segundo creo las variables que voy a necesitar, en este caso como necesito copiar el contenido de un archivo a otro voy a usar el” archivo\_fuente[]” y archivo\_destino[] como cadenas que contienen el nombre del archivo que deseo copiar y donde se guardará la copia.



```
GNU nano 6.2                                copiar_archivo *  
#include <stdio.h>  
#include <stdlib.h>  
  
int main() {  
    char archivo_fuente[] = "archivo_original.txt";  
    //Acá va el nombre del archivo original  
    char archivo_destino[] = "copia_archivo.txt";  
    //aca va el nombre del archivo de destino
```

sourceFile y destFile son punteros a estructuras tipo FILE que representan los archivos originales y los de destino. Y ch es una variable de tipo char para leer y escribir caracteres.



```
FILE *sourceFile, *destFile;  
char ch;
```

sourceFile = fopen(archivo\_fuente, "r"): Abre el archivo original en modo lectura ("r").

destFile = fopen(archivo\_destino, "w"): Crea el archivo de destino en modo escritura ("w").

Se verifica si los archivos se abren correctamente y sinó se muestra un mensaje que dice que no se pudo abrir.

```

//abrir el archivo original en modo lectura
sourceFile = fopen(archivo_fuente, "r");
if (sourceFile == NULL) {
    printf("No se pudo abrir el archivo original.\n");
    return 1;
}

//crear el archivo de destino en modo escritura
destFile = fopen(archivo_destino, "w");
if (destFile == NULL) {
    printf("No se pudo crear el archivo de destino. \n");
    fclose(sourceFile);
    return 1;
}

```

Si se abre el archivo se usa un bucle while para leer cada letra del archivo original (sourceFile) utilizando el fgetc y luego se escribe ese carácter en el archivo de destino destFile utilizando fputc (ch, destFile).

```

//Copiar el contenido del original al archivo de destino
while ((ch = fgetc(sourceFile)) != EOF) {
    fputc(ch, destFile)
}

```

Cierre de Archivos:

Finalmente, se cierran ambos archivos con fclose(sourceFile) y fclose(destFile).

Y si todo funciona correctamente muestra un mensaje de éxito.

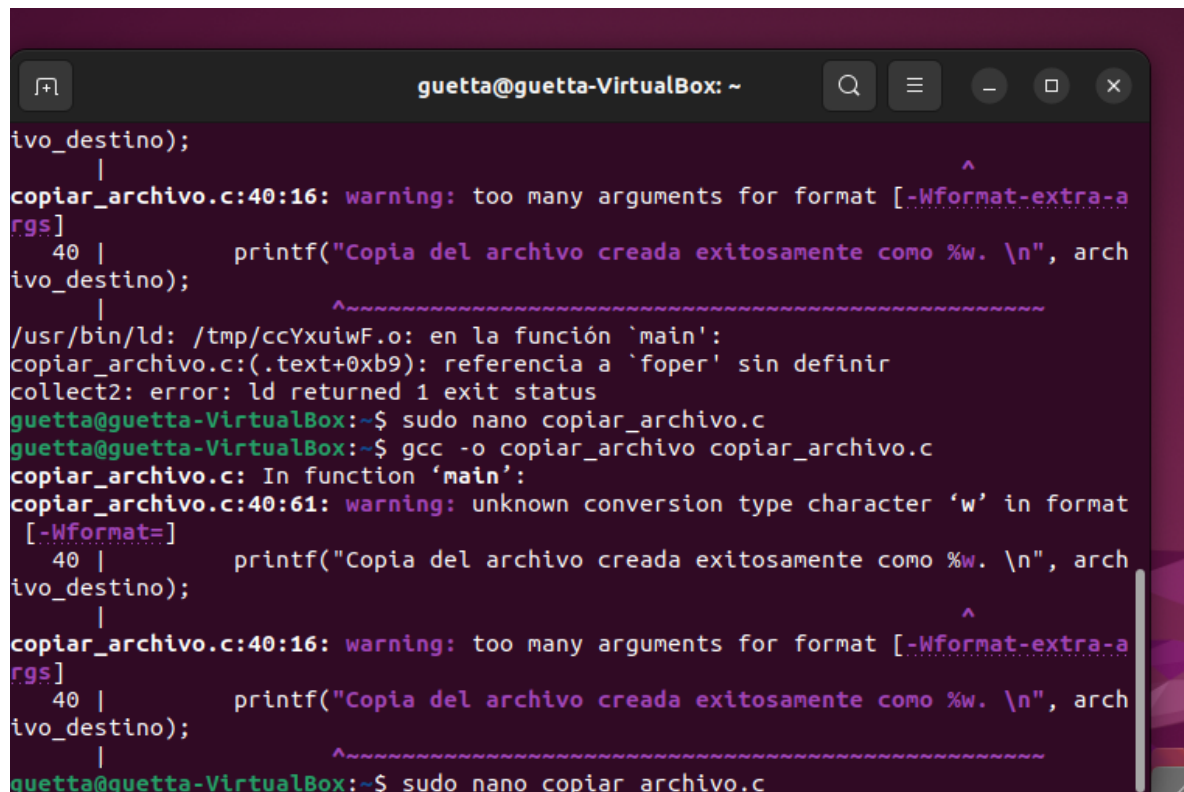
```

fclose(sourceFile);
fclose(destFile);

printf("Copia del archivo creada exitosamente como %w. \n", archivo_destino);
return 0;

```

TOCÓ HACER MUCHAS CORRECCIONES DE ESCRITURA



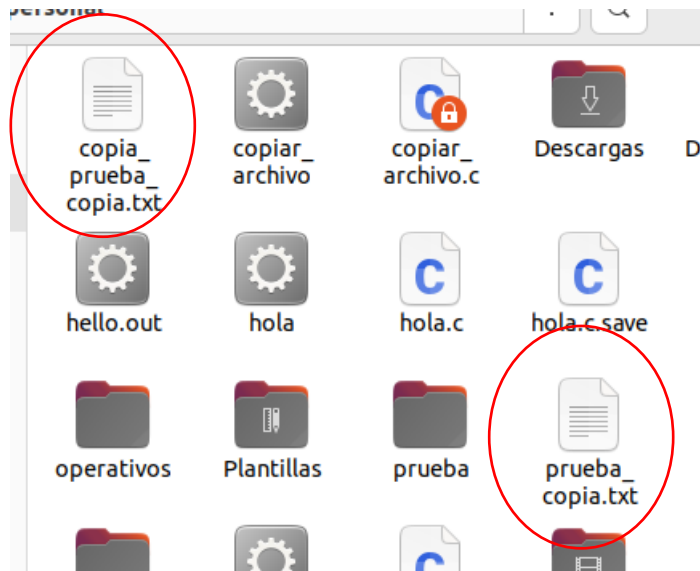
```
guetta@guetta-VirtualBox: ~  
ivo_destino);  
|  
copiar_archivo.c:40:16: warning: too many arguments for format [-Wformat-extra-args]  
40 |         printf("Copia del archivo creada exitosamente como %w. \n", arch  
ivo_destino);  
|  
/usr/bin/ld: /tmp/ccYxuiwF.o: en la función `main':  
copiar_archivo.c:(.text+0xb9): referencia a `foper' sin definir  
collect2: error: ld returned 1 exit status  
guetta@guetta-VirtualBox:~$ sudo nano copiar_archivo.c  
guetta@guetta-VirtualBox:~$ gcc -o copiar_archivo copiar_archivo.c  
copiar_archivo.c: In function `main':  
copiar_archivo.c:40:61: warning: unknown conversion type character 'w' in format  
[-Wformat=]  
40 |         printf("Copia del archivo creada exitosamente como %w. \n", arch  
ivo_destino);  
|  
copiar_archivo.c:40:16: warning: too many arguments for format [-Wformat-extra-args]  
40 |         printf("Copia del archivo creada exitosamente como %w. \n", arch  
ivo_destino);  
|  
guetta@guetta-VirtualBox:~$ sudo nano copiar_archivo.c
```

Hasta que compiló

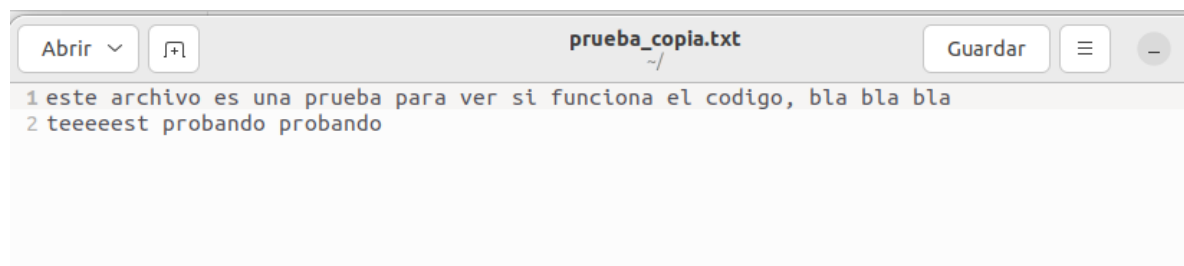
Prueba: Acá se puede observar como se realiza la copia de los el archivo que seleccioné y se guarda como puse.

```
guetta@guetta-VirtualBox:~$ sudo nano copiar_archivo.c
guetta@guetta-VirtualBox:~$ gcc -o copiar_archivo copiar_archivo.c
guetta@guetta-VirtualBox:~$ ./copiar_archivo
Copia del archivo creada exitosamente como copia_prueba_copia.txt.
guetta@guetta-VirtualBox:~$
```

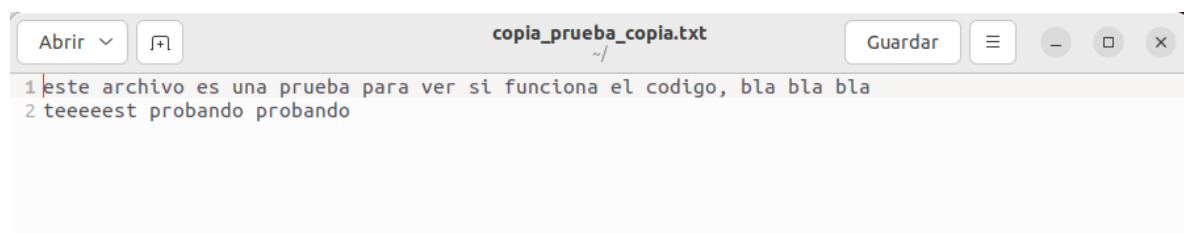
```
] = "prueba_copia.txt";
vo original
[] = "copia_prueba_copia.txt";
vo de destino
```



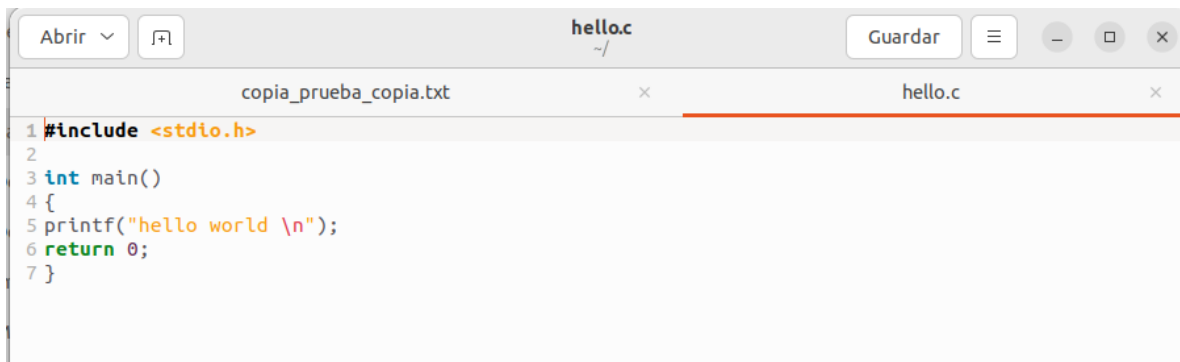
Archivo original:



Copia de archivo:

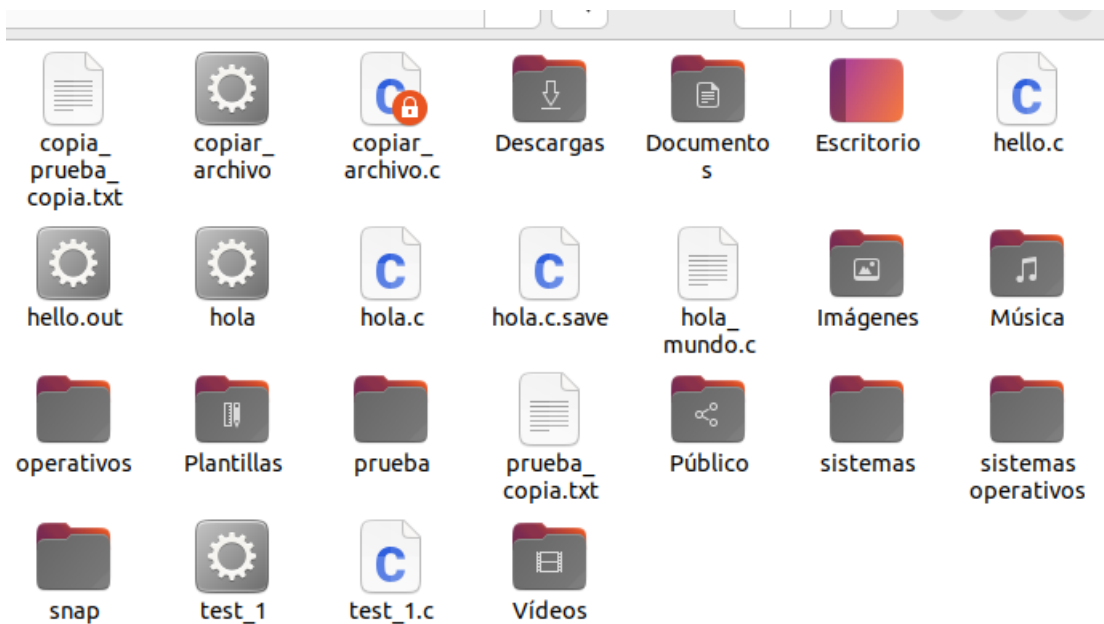


Voy a hacer la prueba con el archivo hello.c



The screenshot shows a code editor window with two tabs: 'copia\_prueba\_copia.txt' and 'hello.c'. The 'hello.c' tab is active, displaying the following C code:

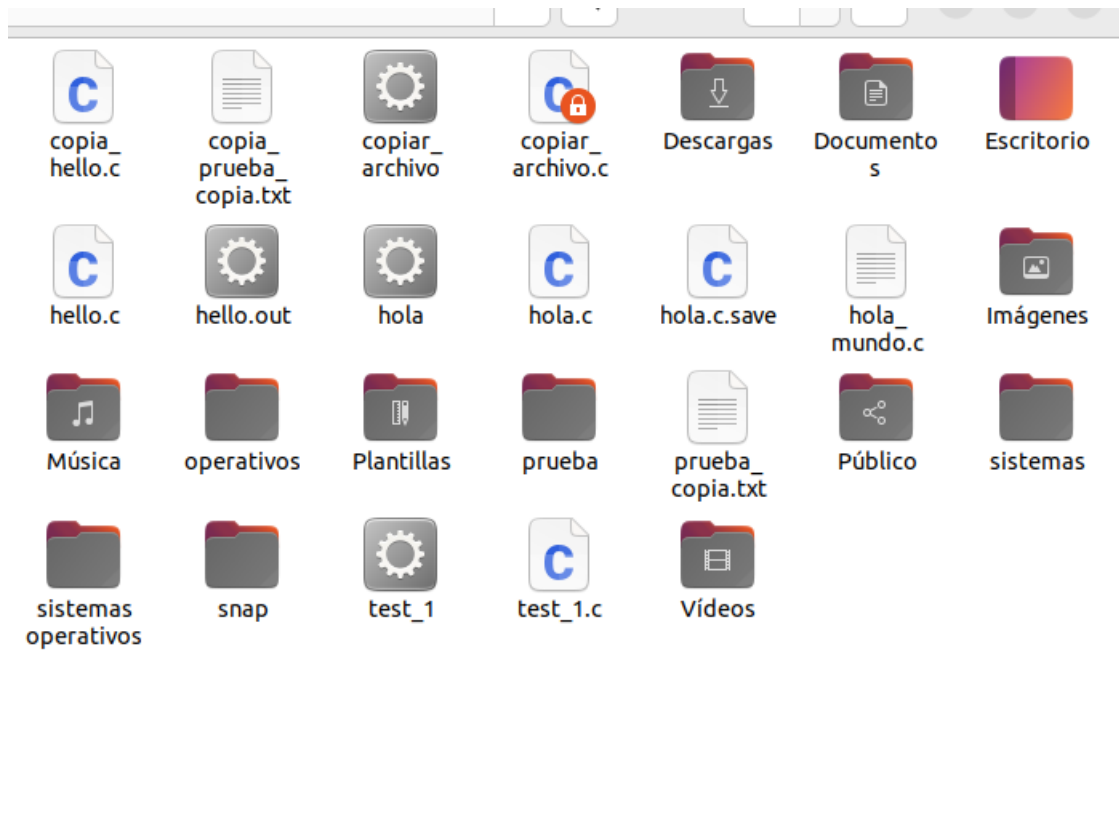
```
1 #include <stdio.h>
2
3 int main()
4 {
5     printf("hello world \n");
6     return 0;
7 }
```



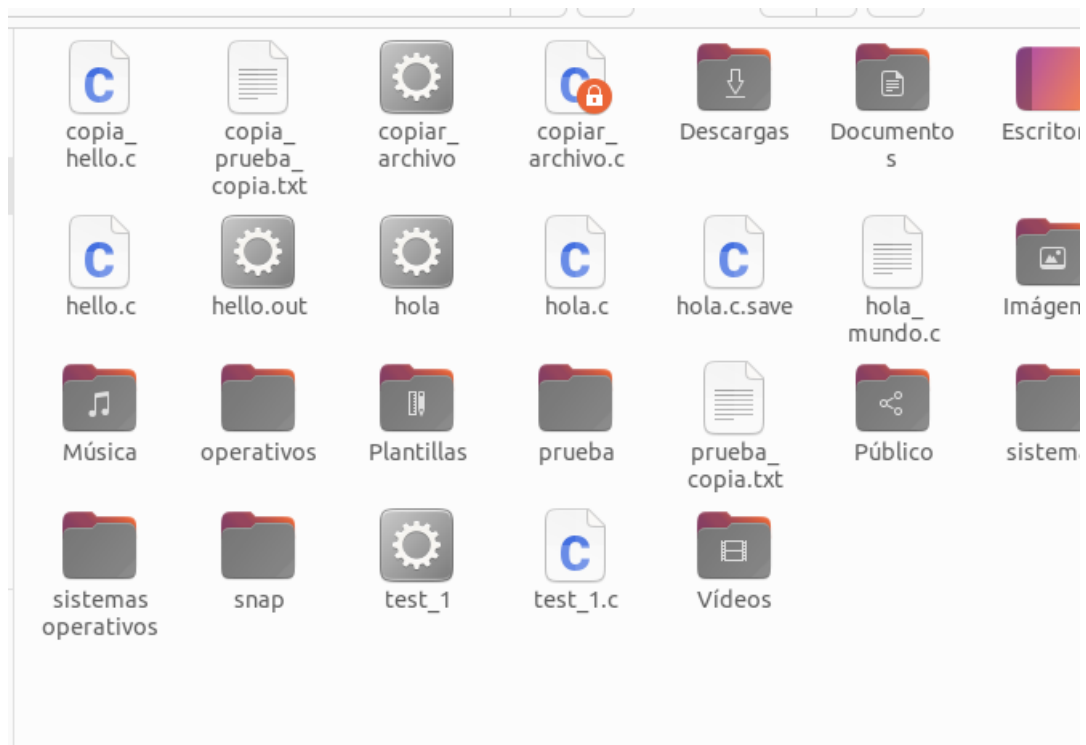
```
= "hello.c";
original
= "copia_hello.c";
de destino
```

```
$ sudo nano copiar_archivo.c
$ sudo nano copiar_archivo.c
$ gcc -o copiar_archivo copiar_archivo.c
$
```

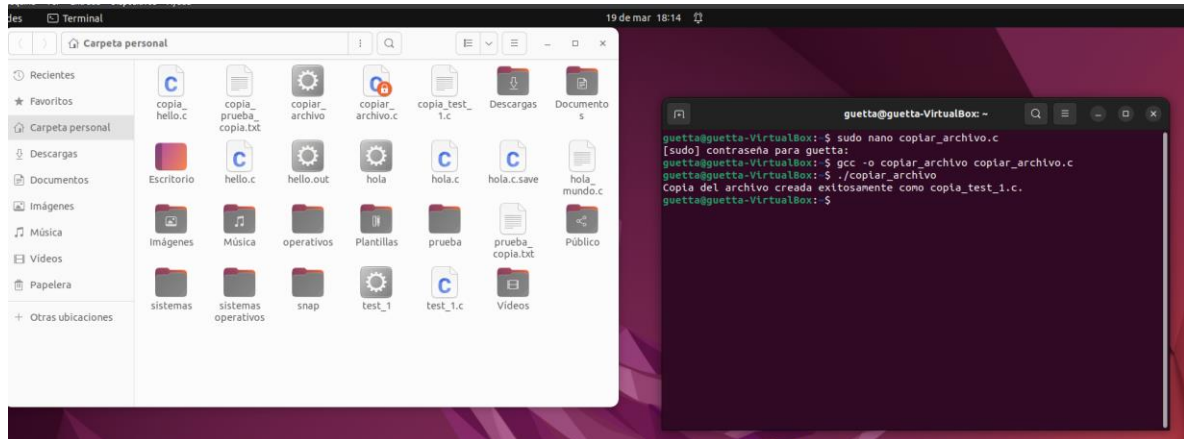
```
guetta@guetta-VirtualBox:~$ ./copiar_archivo
Copia del archivo creada exitosamente como copia_hello.c.
guetta@guetta-VirtualBox:~$
```



Ahora voy a hacer la prueba con el archivo “test\_1.c”



```
vo_fuente[] = "test_1.c";  
del archivo original  
vo_destino[] = "copia_test_1.c";  
del archivo de destino
```



Para copiar los archivos solo hay que entrar al código y escribir el archivo que se quiere copiar y el nombre con el que se quiere copiar.