

1.Crear el archivo fuente en C:

Aquí hemos creado un archivo que contiene el código fuente en C, mediante el editor de texto nano del terminal. Hemos creado el archivo con el siguiente código, lo hemos guardado y hemos cerrado el editor de texto.

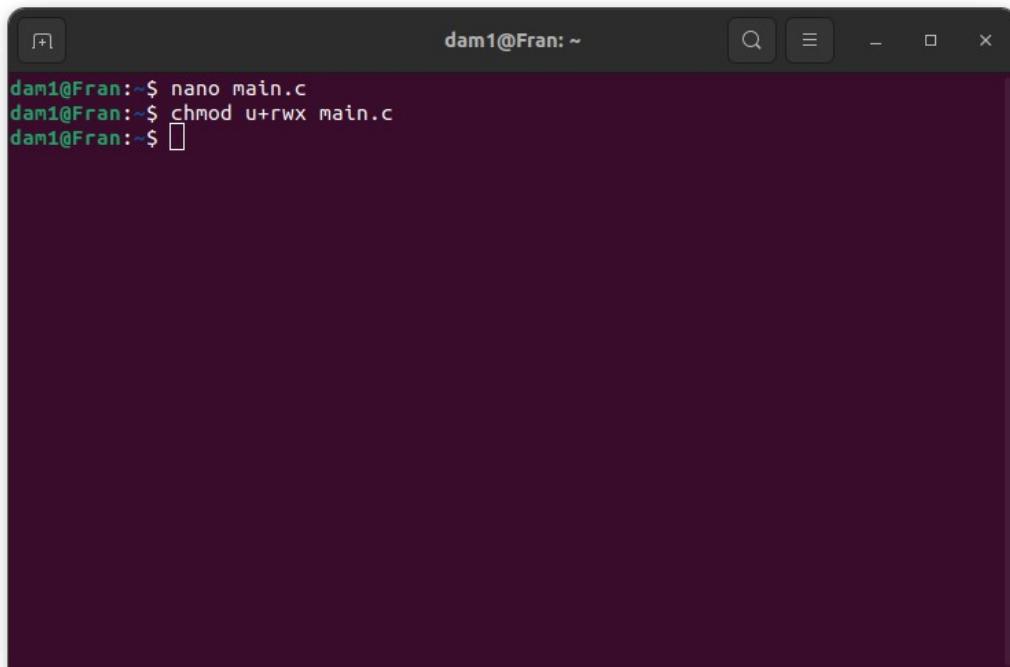
```
#include <stdio.h>
#define MESSAGE "¡Hola estudiantes de DAM!\n"

//los comentarios serán borrados en el preprocesado
//Aquí comienza el programa en sí
int main() {
    printf(MESSAGE);
    return 0;
}
//Aquí finaliza el programa
```



2.Cambiar los permisos del archivo para que sea ejecutable

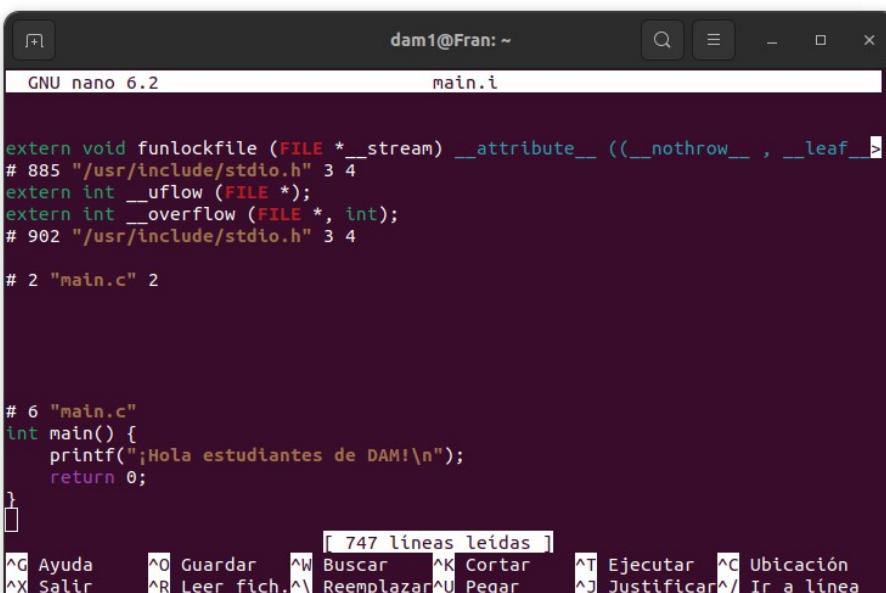
En este paso hemos cambiado los permisos de ejecución mediante el siguiente comando: `chmod u+rwx main.c`



```
dam1@Fran:~$ nano main.c
dam1@Fran:~$ chmod u+rwx main.c
dam1@Fran:~$
```

3.Preprocesado

En este paso vamos a adaptar el código fuente eliminando los comentarios y revisando posibles errores, generando el archivo **main.i** cuyo contenido es el código ya preprocesado.



```
GNU nano 6.2                         main.i

extern void funlockfile (FILE * _stream) __attribute__ ((__nothrow__ , __leaf__))
# 885 "/usr/include/stdio.h" 3 4
extern int __uflow (FILE *);
extern int __overflow (FILE *, int);
# 902 "/usr/include/stdio.h" 3 4

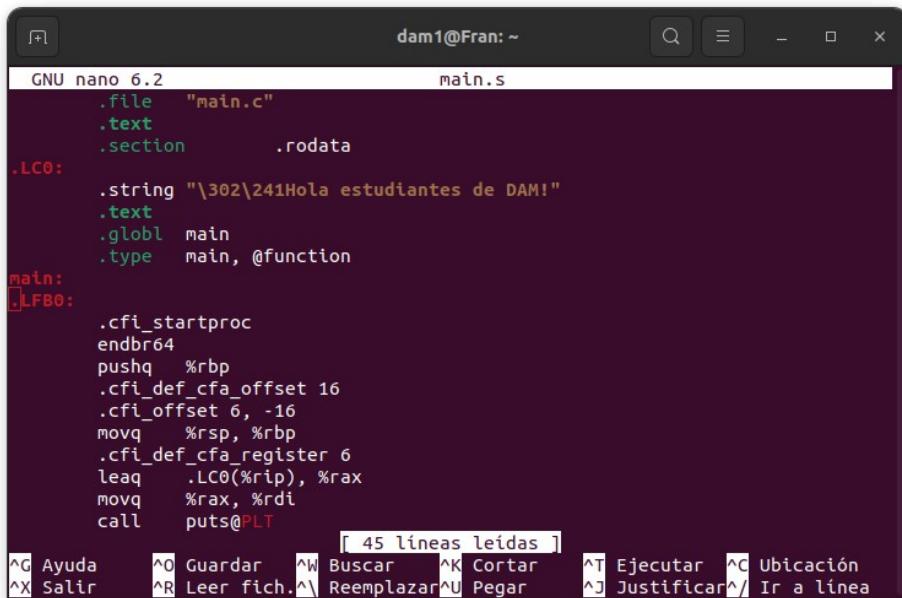
# 2 "main.c" 2

# 6 "main.c"
int main() {
    printf("¡Hola estudiantes de DAM!\n");
    return 0;
}

[ 747 líneas leídas ]
```

^G Ayuda ^O Guardar ^W Buscar ^K Cortar ^T Ejecutar ^C Ubicación
^X Salir ^R Leer fich. ^A Reemplazar ^U Pegar ^J Justificar ^/ Ir a linea

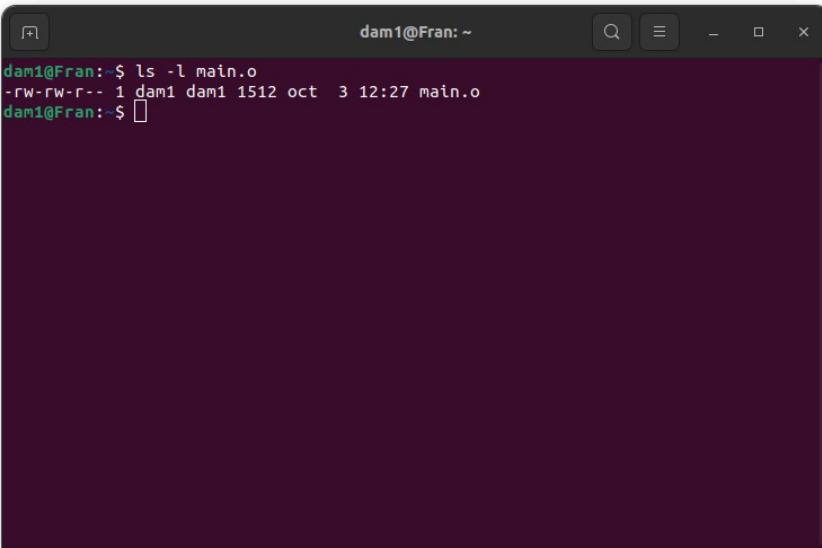
4. Compilación: En este paso vamos a convertir el código procesado en código ensamblador, donde empieza a haber dependencia con el SO y el Software. Aquí generaremos el archivo main.s



The screenshot shows a terminal window titled "dam1@Fran: ~". It displays assembly code for a program named "main.s". The code includes sections like .text and .rodata, and defines symbols such as .LC0 and main. The assembly instructions show the setup of the stack frame and the execution of the main function. At the bottom of the terminal, there is a status bar with keyboard shortcuts for various functions like Ayuda (Help), Guardar (Save), and Ejecutar (Execute).

```
GNU nano 6.2                               main.s
    .file   "main.c"
    .text
    .section      .rodata
.LC0:
    .string "\302\241Hola estudiantes de DAM!"
    .text
    .globl  main
    .type   main, @function
main:
.LFBO:
    .cfi_startproc
    endbr64
    pushq  %rbp
    .cfi_offset %rbp, -16
    movq   %rsp, %rbp
    .cfi_offset %rbp, 6
    .cfi_offset %rbp, 6
    leaq    .LC0(%rip), %rax
    movq   %rax, %rdi
    call   puts@PLT
[ 45 líneas leidas ]
^G Ayuda      ^O Guardar     ^W Buscar     ^K Cortar     ^T Ejecutar     ^C Ubicación
^X Salir      ^R Leer fich.  ^\ Reemplazar ^U Pegar      ^J Justificar ^/ Ir a linea
```

5. Ensamblado: Aquí se va a traducir el código ensamblador a un archivo objeto(binario intermedio), y vamos a generar el archivo **main.o**.



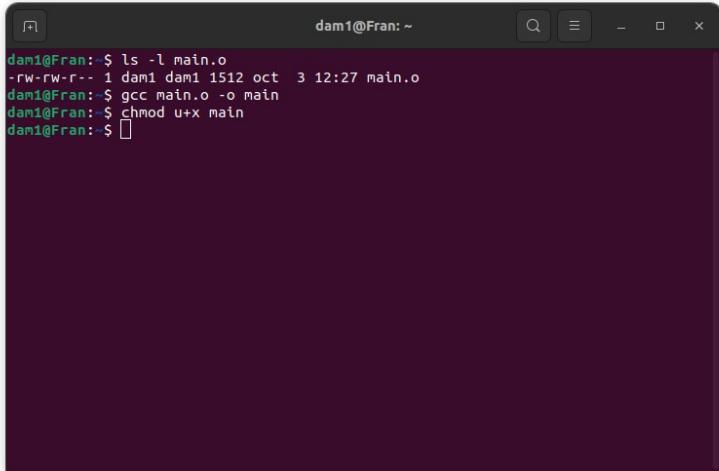
The screenshot shows a terminal window titled "dam1@Fran: ~". It displays the output of the "ls -l" command, which lists a file named "main.o" with the following permissions: -rwx-rw-r--. The file was created by user "dam1" on October 3 at 12:27. There is also a small empty square icon next to the file name.

```
dam1@Fran:~$ ls -l main.o
-rw-rw-r-- 1 dam1 dam1 1512 oct  3 12:27 main.o
dam1@Fran:~$
```

Aquí podemos observar que el código no está en binario por que Linux no lo ha sabido interpretar

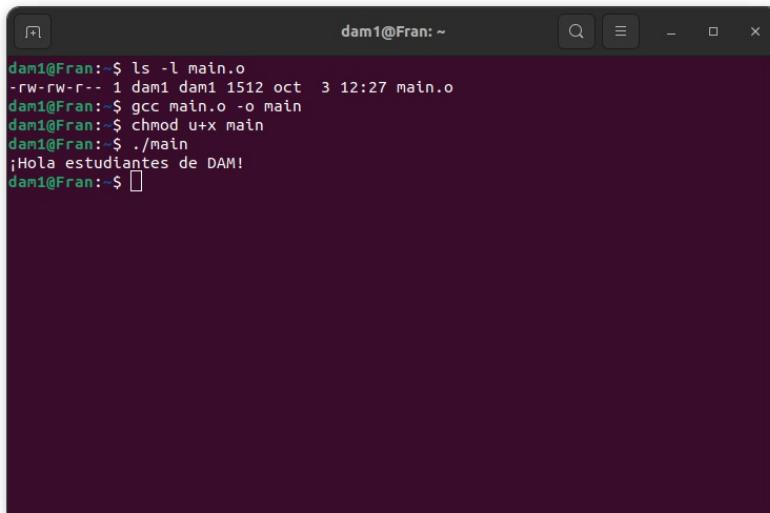
para poder ver el código en binario debemos introducir el siguiente código en la terminal: **`xxd -b main`**

6. Enlazado: Este es el último paso antes de la ejecución, donde enlazaremos las librerías



```
dam1@Fran:~$ ls -l main.o
-rw-rw-r-- 1 dam1 dam1 1512 oct  3 12:27 main.o
dam1@Fran:~$ gcc main.o -o main
dam1@Fran:~$ chmod u+x main
dam1@Fran:~$ 
```

7. Ejecutar el programa: En este paso ejecutaremos el programa a través del siguiente comando: ./main



```
dam1@Fran:~$ ls -l main.o
-rw-rw-r-- 1 dam1 dam1 1512 oct  3 12:27 main.o
dam1@Fran:~$ gcc main.o -o main
dam1@Fran:~$ chmod u+x main
dam1@Fran:~$ ./main
;Hola estudiantes de DAM!
dam1@Fran:~$ 
```