

Práctica 1, llamadas al sistema



Javier Bautista Rosell

NIA: 100315139

Correo: 100315139@alumnos.uc3m.es

Práctica 1, Llamadas al sistema

Mydiff:.....	2
Batería de pruebas:	2
Conclusión:.....	3
Myhead:	4
Batería de pruebas:	4
Conclusión:.....	5
Myls:.....	6
Batería de pruebas:	6
Conclusión:.....	6
Prueba de formato con format.sh.....	7

Mydiff:

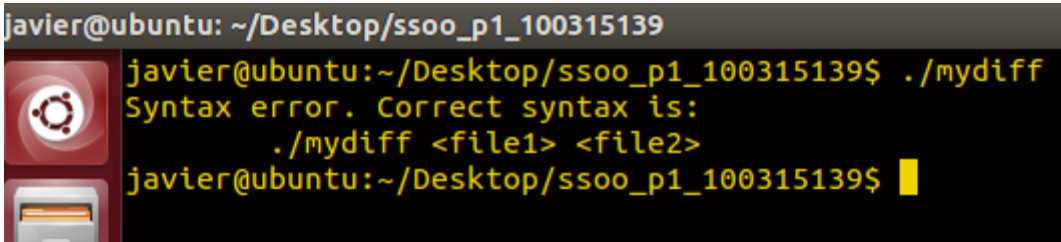
La función principal del programa es comparar dos archivos y decir si son iguales o diferentes, para ello sigue estos pasos:

1. Comprueba que los argumentos introducidos son correctos, en caso contrario devuelve un texto por pantalla indicando la forma correcta de añadirlo, el programa acaba.
2. Abre el archivo que es dado por el primer argumento y comprueba que no falla, en caso de error el programa devuelve "no se puede abrir el fichero 1", el programa acaba.
3. Abre el archivo que es dado por el segundo argumento y comprueba que no falla, en caso de error el programa devuelve "no se puede abrir el fichero 1", cierra el archivo 1, comprueba que cierra bien el programa acaba.
4. Lee los archivos y los compara, en caso de que sean iguales el bucle llega al final y el programa devuelve "los ficheros son iguales", en caso de que no lo sean en cuanto cambien el programa sale del bucle y devuelve "los ficheros son diferentes".

Mientras dura la lectura se comprueba que se lean los ficheros si hay algún fallo de lectura el programa devuelve un error "error al leer el fichero x" dependiendo del fichero que haya dado el fallo, se acaba el programa.

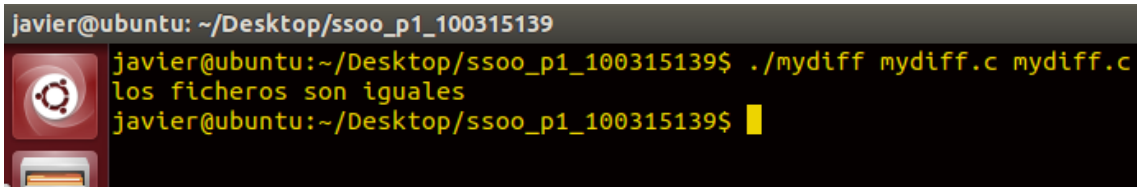
Batería de pruebas:

Caso de no poner los argumentos:

A terminal window with a dark background and yellow text. The prompt is 'javier@ubuntu: ~/Desktop/ssoo_p1_100315139'. The user enters './mydiff'. The output is 'Syntax error. Correct syntax is: ./mydiff <file1> <file2>'. The prompt returns.

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./mydiff
Syntax error. Correct syntax is:
    ./mydiff <file1> <file2>
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Caso de poner el mismo archivo como ambos argumentos:

A terminal window with a dark background and yellow text. The prompt is 'javier@ubuntu: ~/Desktop/ssoo_p1_100315139'. The user enters './mydiff mydiff.c mydiff.c'. The output is 'los ficheros son iguales'. The prompt returns.

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./mydiff mydiff.c mydiff.c
los ficheros son iguales
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Caso de poner 2 archivos que sean diferentes como argumentos:

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./mydiff mydiff.c myls.c
los fiheros son diferentes
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Caso 2 ficheros iguales con distinto nombre y en distinta carpeta:

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./mydiff mydiff.c /home/javier/Desktop/prueba_igual.c
los ficheros son iguales
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Caso error al abrir fichero:

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./mydiff mydiff.c /home/javier/Desktop/archivonulo
no se puede abrir el fichero 2
: Permission denied
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Conclusión:

No encontré mayor problema al realizar este código que un fallo mío al programar la parte del read, que di mal los valores y no me funcionaba hasta que me di cuenta del error, por lo demás era un programa no demasiado complejo.

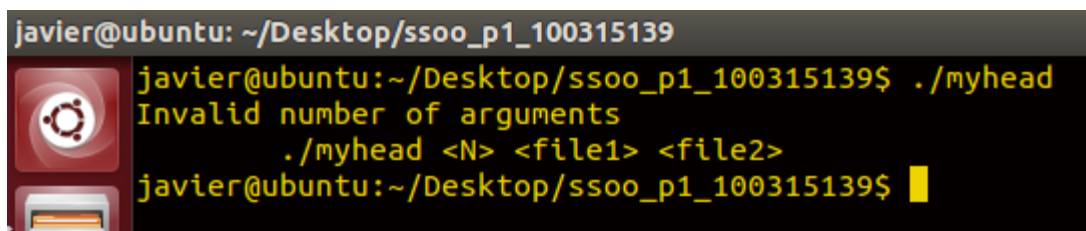
Myhead:

La función de este programa es copiar un número de bits de un archivo a otro, en caso de existir el segundo archivo se borra lo escrito en él y se escribe lo nuevo, en caso de no existir se crea este archivo y se escribe, posteriormente se lee el segundo archivo, estos son los pasos que sigue:

1. Mira a ver que el número de argumentos sea el correcto.
2. Abre el segundo argumento dado (el primero es el número de bits que quieres copiar) y comprueba que no da fallo, en caso de fallo se acaba el programa y devuelve por pantalla que no ha podido ser abierto el fichero.
3. Abre el tercer argumento, en caso de no existir se crea el archivo, comprueba que no da error al abrirlo, en caso de error cierra el fichero anterior y acaba el programa comunicándolo por pantalla.
4. Crea un contador y una variable que indica el número de bits que se desean copiar
5. Se hace un bucle que lee del primer fichero y lo escribe en el segundo a menos que de error al escribir o al leer en alguno, si se acaba de leer el primer fichero el bucle se detiene, en caso de haber copiado los bits pedidos también.
6. Cierra ambos ficheros
7. Vuelve a abrir el fichero 2 para leerlo, realiza todas las comprobaciones de antes y lo mete en un bucle de lectura, al acabar el bucle se cierra el fichero de nuevo.

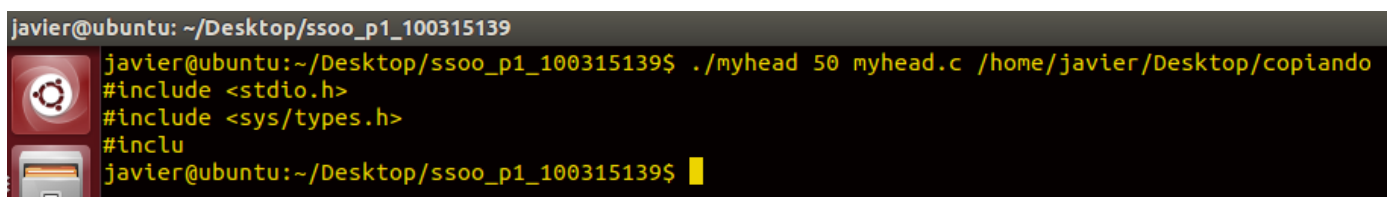
Batería de pruebas:

Caso de no poner argumentos:

A terminal window with a dark background and yellow text. The prompt is 'javier@ubuntu: ~/Desktop/ssoo_p1_100315139'. The user enters './myhead', and the program outputs 'Invalid number of arguments' followed by a new line and the usage instructions './myhead <N> <file1> <file2>'. The prompt returns to 'javier@ubuntu: ~/Desktop/ssoo_p1_100315139\$'.

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./myhead
Invalid number of arguments
./myhead <N> <file1> <file2>
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

No existe el segundo fichero, se crea:

A terminal window with a dark background and yellow text. The prompt is 'javier@ubuntu: ~/Desktop/ssoo_p1_100315139'. The user enters './myhead 50 myhead.c /home/javier/Desktop/copiando'. The program outputs '#include <stdio.h>', '#include <sys/types.h>', and '#inclu'. The prompt returns to 'javier@ubuntu: ~/Desktop/ssoo_p1_100315139\$'.

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./myhead 50 myhead.c /home/javier/Desktop/copiando
#include <stdio.h>
#include <sys/types.h>
#inclu
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Caso de no poder abrir un fichero:

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./myhead 500 myhead.c /home/javier/Desktop/archivonulo
no se puede abrir el fichero 2
: Permission denied
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Caso fichero ya creado

```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./myhead 5 myhead.c /home/javier/Desktop/copiando
#incl
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Conclusión:

Me daba problemas el hecho de abrir el segundo fichero y dejarlo abierto para leerlo posteriormente por lo que he tenido que arreglarlo cerrándolo y volviéndolo a abrir, y eso ha hecho que el programa pueda funcionar, en caso de haberlo dejado como estaba me daría errores de lectura todo el rato del fichero 2.

También me daba problema al crear el fichero pero era porque no le había dado permisos.

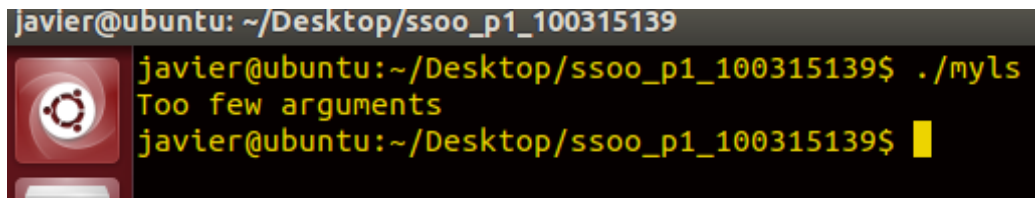
Myls:

El programa realiza la misma función que si aplicas en el terminal de Linux **"ls -lf"** es decir te lista la dirección que le das, para ello el programa sigue los siguientes pasos:

1. Comprueba el número de argumentos.
2. Utiliza la función opendir para abrir el directorio que le pasamos por el argumento 1
3. Comprueba que se haya abierto es decir dir!=null.
4. Realiza un bucle que recorra las posiciones de memoria del directorio imprimiendo los nombres de los diferentes archivos que están en él.
5. Cierra el directorio.

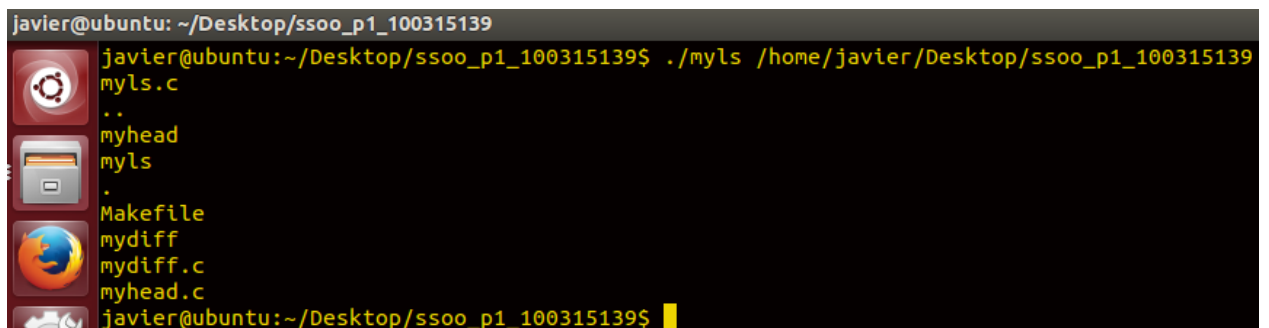
Batería de pruebas:

Caso de no poner suficientes argumentos:



```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./mys
Too few arguments
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Caso de que se ejecute correctamente:



```
javier@ubuntu: ~/Desktop/ssoo_p1_100315139
javier@ubuntu:~/Desktop/ssoo_p1_100315139$ ./mys /home/javier/Desktop/ssoo_p1_100315139
mys.c
..
myhead
mys
.
Makefile
mydiff
mydiff.c
myhead.c
javier@ubuntu:~/Desktop/ssoo_p1_100315139$
```

Conclusión:

El mayor problema con esta práctica ha sido el uso de opendir y encontrar d_name para poder listar los ficheros, por lo demás es un código bastante más corto que los dos anteriores.

Prueba de formato con format.sh

```
a0315139@guernika:~$ ./format.sh ss00_p1_100315139.zip
ZIP name/format OK
Archive:  ss00_p1_100315139.zip
  inflating: unzip/Makefile
  inflating: unzip/mydiff.c
  inflating: unzip/myhead.c
  inflating: unzip/myls.c
myls.c OK
Makefile OK
LS Compilation OK
LS Warnings OK
LS OK
```