

Programación y Administración de Sistemas

Práctica 1. Programación de *shell scripts*

Convocatoria de junio (curso 2017/2018)

Manuel Dorado Moreno

7 de marzo de 2018

Resumen

Se entregará el código de los *scripts*, junto con un fichero de texto que aclare las particularidades de los programas desarrollados e incluya ejemplos de ejecución. Es obligatorio comprobar errores en los argumentos que reciben los *scripts*. Se valorará la utilización de comentarios, la máxima modularidad en el código y la claridad en las salidas generadas. Todos los *scripts* deben prepararse para funcionar correctamente con el intérprete de órdenes `bash` de la máquina `ts.uco.es`. La entrega se hará utilizando la tarea en Moodle habilitada al efecto. En caso de que dos alumnos entreguen códigos copiados, no se puntuarán ninguno de los dos. Comprueba que los comportamientos de los programas son similares a los esperados en los ejemplos de ejecución. Debes respetar los nombres de los *scripts* (`ejercicio1.sh...`). La fecha de entrega de la práctica será el próximo **domingo 4 de marzo a las 23:55h**.

1. `ejercicio1.sh`

Desarrollar un *script* que reciba un único argumento que será un directorio. En primer lugar, contar cuantos ficheros hay con extensión `.c` y cuantos con extensión `.h`. Además, el *script* debe mostrar la lista de ficheros con extensión `.c` o `.h` contenidos en ese directorio, ordenados según el número de caracteres de cada fichero (de mayor a menor) e indicar también en el listado el número de líneas y de caracteres. Un ejemplo de ejecución es el siguiente:

```
1 mdorado@mdoradoLaptop:~/pas/1718$ ./ejercicio1.sh ejemploCarpeta/  
2 Tenemos 4 ficheros con extensión .c y 5 ficheros con extensión .h  
3 El fichero ejemploCarpeta/d1/audit.h tiene 463 líneas y 18221 caracteres  
4 El fichero ejemploCarpeta/d1/auditBackup.h tiene 463 líneas y 18221 caracteres  
5 El fichero ejemploCarpeta/d1/ecoc.h tiene 76 líneas y 2377 caracteres  
6 El fichero ejemploCarpeta/d1/auto_fs.h tiene 72 líneas y 2288 caracteres  
7 El fichero ejemploCarpeta/d1/compat.h tiene 17 líneas y 370 caracteres  
8 El fichero ejemploCarpeta/d2/prueba.c tiene 14 líneas y 233 caracteres  
9 El fichero ejemploCarpeta/d1/smo_routine.c tiene 8 líneas y 104 caracteres  
10 El fichero ejemploCarpeta/d1/svm_model_matlab.c tiene 6 líneas y 99 caracteres  
11 El fichero ejemploCarpeta/d1/d4/act_func.c tiene 5 líneas y 92 caracteres
```

2. `ejercicio2.sh`

Crear un *script* que reciba como argumento un parámetro que será un directorio y un segundo argumento que será un número (número de *bytes*). El *script* debe buscar todos los ficheros que estén alojados en la carpeta que se pasa como primer argumento, cuyo tamaño total en bytes sea mayor o igual que el número pasado como segundo argumento. Si no se especifica un segundo argumento, el *script* deberá mostrar todos los ficheros (es decir, como si el tamaño fuese 0). Para cada fichero, habrá que mostrar:

1. El nombre del fichero (sin la ruta).

2. El número de caracteres del nombre del usuario propietario.
3. La fecha de creación (legible) y del último acceso al fichero (segundos UNIX).
4. Tamaño en bytes y número de bloques que ocupa en memoria
5. La cadena de permisos del fichero
6. Un 1 si el fichero es ejecutable o un 0 si no lo es.

Todos estos campos se separarán por el carácter “;” y la salida deberá ordenarse numéricamente por tamaño (de menor a mayor). Un ejemplo de ejecución del *script* es el que aparece a continuación:

```

1 mdoorado@mdoradoLaptop:~/pas/1718$ ./ejercicio2.sh ejemploCarpeta
2 Nombre,LongitudUsuario,FechaModificacion,FechaAcceso,Tamano,Bloques,Permisos,Ejecutable
3 act_func.c;7;2016-02-28 18:09:30.000000000 +0100;1518090342;92;8;-rw-----,0
4 svm_model_matlab.c;7;2016-02-28 18:09:24.000000000 +0100;1518090342;99;8;-rw-----,0
5 smo_routine.c;7;2016-02-28 18:09:21.000000000 +0100;1518090342;104;8;-rw-----,0
6 prueba.c;7;2016-02-28 18:08:56.000000000 +0100;1518090342;233;8;-rw-----,0
7 compat.h;7;2016-02-28 18:09:22.000000000 +0100;1518090342;370;8;-rw-----,0
8 ejemploSuid.sh;7;2016-02-28 18:08:57.000000000 +0100;1518090342;483;8;-rw-----,0
9 bcache.cpp;7;2016-02-28 18:09:31.000000000 +0100;1518090342;1233;8;-rw-----,0
10 auto_fs.h;7;2016-02-28 18:09:26.000000000 +0100;1518090342;2288;8;-rw-----,0
11 ecoc.h;7;2016-02-28 18:09:23.000000000 +0100;1518090342;2377;8;-rw-----,0
12 fedora.png;7;2016-02-28 18:09:17.000000000 +0100;1518090342;2386;8;-rw-----,0
13 debian.png;7;2016-02-28 18:09:16.000000000 +0100;1518090342;2626;8;-rw-----,0
14 a2.out;7;2016-02-28 18:09:13.000000000 +0100;1518090342;7275;16;-rwx-----,1
15 a.out;7;2016-02-28 18:08:55.000000000 +0100;1518090342;7275;16;-rwx-----,1
16 eje2;7;2016-02-28 18:09:34.000000000 +0100;1518090342;7275;16;-rwx-----,1
17 centos.png;7;2016-02-28 18:09:16.000000000 +0100;1518090342;8215;24;-rw-----,0
18 pirates.jpg;7;2016-02-28 18:09:06.000000000 +0100;1518090342;9836;24;-rw-----,0
19 libhandle.so.1.0.3;7;2016-02-28 18:09:25.000000000 +0100;1518090342;10760;24;-rw-----,0
20 Linus_torvalds.jpg;7;2016-02-28 18:09:04.000000000 +0100;1518090342;17670;40;-rw-----,0
21 auditBackup.h;7;2017-02-24 11:46:53.000000000 +0100;1518090342;18221;40;-rw-----,0
22 audit.h;7;2017-02-24 11:46:02.000000000 +0100;1518091627;18221;40;-rw-----,0
23 Logo_uco.gif;7;2016-02-28 18:09:10.000000000 +0100;1518090342;20005;40;-rw-----,0
24 devocion-administrador-de-sistemas.png;7;2016-02-28 18:09:07.000000000 +0100;1518090342;21613;48;-rw
25 -----,0
26 HD.png;7;2016-02-28 18:09:14.000000000 +0100;1518090342;36710;72;-rw-----,0
27 libsysfs.so.2;7;2016-02-28 18:09:29.000000000 +0100;1518090342;38644;80;-rw-----,0
28 richard-stallman.jpg;7;2016-02-28 18:09:05.000000000 +0100;1518090342;42999;88;-rw-----,0
29 logo2.jpg;7;2016-02-28 18:09:10.000000000 +0100;1518090342;43161;88;-rw-----,0
30 logo_eps_grande.jpg;7;2016-02-28 18:09:10.000000000 +0100;1518090342;43161;88;-rw-----,0
31 raid0.png;7;2016-02-28 18:09:18.000000000 +0100;1518090342;50683;104;-rw-----,0
32 elinfiernoexiste.jpg;7;2016-02-28 18:09:02.000000000 +0100;1518090342;119847;240;-rw-----,0
33 libglib-2.0.so.0;7;2016-02-28 18:08:55.000000000 +0100;1518090342;822344;1608;-rw-----,0
34
35
36 mdoorado@mdoradoLaptop:~/pas/1718$ ./ejercicio2.sh ejemploCarpeta 5000
37 Nombre,LongitudUsuario,FechaModificacion,FechaAcceso,Tamano,Bloques,Permisos,Ejecutable
38 a2.out;7;2016-02-28 18:09:13.000000000 +0100;1518090342;7275;16;-rwx-----,1
39 a.out;7;2016-02-28 18:08:55.000000000 +0100;1518090342;7275;16;-rwx-----,1
40 ...

```

Si redirigimos la salida producida hacia un archivo .csv, ésta se debería poder abrir utilizando LibreOffice Calc (habría que elegir como separador el punto y coma):

```

1 mdoorado@mdoradoLaptop:~/pas/1718$ ./ejercicio2.sh ejemploCarpeta/ 5000 > salida.csv
2 mdoorado@mdoradoLaptop:~/pas/1718$ libreoffice5.0 salida.csv

```

3. ejercicio3.sh

Crear un *script* que clasifique los ficheros que hay dentro de una carpeta en función de su tamaño en pequeños, medianos y grandes. El primer argumento del *script* recibido por línea de comandos será la carpeta a procesar (siendo un argumento obligatorio). Habrá dos argumentos adicionales y opcionales que serán los umbrales para discernir las tres categorías (*umbral1* y *umbral2*), de forma que los ficheros pequeños serán todos aquellos cuyo tamaño sea menor que

umbral1, los medianos serán aquellos ficheros cuyo tamaño esté entre umbral1 y umbral2 (incluidos), y los grandes serán los ficheros con tamaño mayor que umbral2. Se supone que los tamaños se especifican en *bytes*. El valor por defecto para umbral1 y umbral2 será 10000 y 100000, respectivamente. El *script* debe crear tres subcarpetas (pequenos, medianos y grandes) dentro de la carpeta en la que se realiza la llamada al *script* e incluir en cada una los ficheros correspondientes. Si las carpetas ya existen, borrarlas.

Un ejemplo de ejecución es el siguiente (observa como el fichero libhandle.so.1.0.3 no es “mediano” en la segunda ejecución):

```

1 mdorado@mdoradoLaptop:~/pas/1718$ ./ejercicio3.sh ejemploCarpeta/
2 Creando las carpetas pequenos, medianos y grandes...
3 Copiando los archivos...
4 mdorado@mdoradoLaptop:~/pas/1718$ ls pequenos/ medianos/ grandes/
5 grandes/:
6 elinfiernoexiste.jpg libglib-2.0.so.0
7 medianos/:
8 audit.h                                HD.png                                libsysfs.so.2                        logo_eps_grande.jpg
9 raid0.png
10 devocion-administrador-de-sistemas.png libhandle.so.1.0.3 Linus_torvalds.jpg Logo_uco.gif
11 richard-stallman.jpg
12 pequenos/:
13 a2.out          a.out          bcache.cpp    compat.h      ecoc.h        ejemploSuid.sh  pirates.jpg    smo_routine.c
14 act_func.c      auto_fs.h      centos.png    debian.png    eje2          Fedora.png      prueba.c       svm_model_matlab.
15 c
16
17 mdorado@mdoradoLaptop:~/pas/1718$ ls -la medianos/libhandle.so.1.0.3
18 -rw----- 1 i02gupep upi0 10760 feb 28 13:26 medianos/libhandle.so.1.0.3
19
20 mdorado@mdoradoLaptop:~/pas/1718$ ./ejercicio3.sh ejemploCarpeta/ 10761 100000
21 Las carpetas de salida ya existen, se va a proceder a borrarlas...
22 Copiando los archivos...
23 mdorado@mdoradoLaptop:~/pas/1718$ ls pequenos/ medianos/ grandes/
24 grandes/:
25 elinfiernoexiste.jpg libglib-2.0.so.0
26 medianos/:
27 audit.h devocion-administrador-de-sistemas.png HD.png libsysfs.so.2 Linus_torvalds.jpg
28 logo_eps_grande.jpg Logo_uco.gif raid0.png richard-stallman.jpg
29 pequenos/:
30 a2.out          a.out          bcache.cpp    compat.h      ecoc.h        ejemploSuid.sh  libhandle.so.1.0.3  prueba.c
31 svm_model_matlab.c
32 act_func.c      auto_fs.h      centos.png    debian.png    eje2          Fedora.png      pirates.jpg
33 smo_routine.c

```

4. ejercicio4.sh

Escribir un *script* que realice las siguientes acciones¹:

- Mostrar el número de archivos en el directorio actual, sin contar directorios.
- Mostrar una lista ordenada de los usuarios que hay *logeados* en el sistema en el instante en que se ejecuta el *script*, eliminando los usuarios repetidos².
- Imprimir el número de veces que aparece un determinado carácter, que se le solicita al usuario, en nombres de ficheros contenidos en la carpeta actual o en alguna subcarpeta. Si el usuario no indica el carácter en 5 segundos, utilizar la letra 'a'.

A continuación se muestra un ejemplo de ejecución:

```

1 mdorado@mdoradoLaptop ~/pas1718 $ cd ejemploCarpeta/
2 mdorado@mdoradoLaptop ~/pas1718/ejemploCarpeta $ ../ejercicio4.sh
3 El número de archivos en la carpeta actual es 0
4 mdorado
5 i02gupep
6 javier
7 ¿Qué carácter quieres contar? d

```

¹Puedes necesitar los comandos `wc` (para contar) y `tr` (para reemplazar caracteres)

²Consultar los comandos `who` y `uniq`

```

8 | El carácter d aparece 20 veces en nombres de ficheros o carpetas contenidos en la carpeta actual
9 | mdorado@mdoradoLaptop: ~/pas1718/ejemploCarpeta $
10 | mdorado@mdoradoLaptop: ~/pas1718/ejemploCarpeta $ cd dl/
11 | mdorado@mdoradoLaptop: ~/pas1718/ejemploCarpeta/dl $ ../../ejercicio4.sh
12 | El número de archivos en la carpeta actual es 8
13 | mdorado
14 | i02gupep
15 | javier
16 | ¿Qué carácter quieres contar? T
17 | El carácter T aparece 0 veces en nombres de ficheros o carpetas contenidos en la carpeta actual

```

5. ejercicio5.sh

Realizar un *script* que aplique una copia de seguridad de ficheros o directorios. Todos los ficheros o directorios que se pasen por argumento en la línea de comandos al *script* se comprimirán juntos en un único fichero `copia-usuario-fecha.tar.gz`, donde `usuario` es el usuario que está ejecutando el *script* y `fecha` es la fecha en segundos desde el 1 de enero de 1970. Una vez comprimidos, los archivos se moverán a la carpeta `~/Copia`. Si la carpeta no existe, hay que crearla. Al invocar el *script*, todos los ficheros de la carpeta `~/Copia` con una antigüedad mayor a 200 segundos, deberán ser borrados. Un ejemplo de ejecución es el siguiente:

```

1 | mdorado@mdoradoLaptop:~/pas/1718$ ./ejercicio5.sh carpetaPrueba/
2 | La copia de seguridad se ha creado correctamente.
3 |
4 | mdorado@mdoradoLaptop:~/pas/1718$ ls ~/Copia/ -la
5 | total 720
6 | drwx----- 2 mdorado upi0      80 feb 24 13:52 .
7 | drwx--x--x 70 mdorado upi0    6144 feb 24 13:52 ..
8 | -rw----- 1 mdorado upi0 719039 feb 24 13:52 copia-mdorado-1487940763.tar.gz
9 |
10 | mdorado@mdoradoLaptop:~/pas/1718$ ./ejercicio5.sh carpetaPrueba/
11 | La copia de seguridad se ha creado correctamente.
12 | Borrando /home/mdorado/Copia/copia-mdorado-1487940763.tar.gz de 205 segundos...
13 | La copia de seguridad se ha creado correctamente.

```

6. ejercicio6.sh

Realizar un *script* que reciba como argumento el nombre de un directorio y genera un fichero `.html` con el listado de ficheros y/o carpetas de dicho directorio. El *script* deberá navegar recursivamente por todas las carpetas que haya incluidas en el directorio. Las carpetas deberán representarse en negrita, mientras que los ficheros en tipo de texto normal. Para agrupar el contenido de cada carpeta utilizaremos una enumeración (``) que estará compuesta de un conjunto de elementos (``). Para que las carpetas aparezcan en negrita puedes utilizar la etiqueta ``. El nombre del fichero `.html` será `dir.html`, donde `dir` es el nombre de la carpeta. Un ejemplo de ejecución es el siguiente:

```

1 | mdorado@mdoradoLaptop:~/pas/1718$ ./ejercicio6.sh .
2 | Generando el listado de la carpeta /home/mdorado/pas/1718 sobre el fichero 1617.html...
3 | ¡Terminado!
4 | mdorado@mdoradoLaptop:~/pas/1718$ ./ejercicio6.sh ejemploCarpeta/
5 | Generando el listado de la carpeta /home/mdorado/pas/1718/ejemploCarpeta sobre el fichero
   | ejemploCarpeta.html...
6 | ¡Terminado!
7 | mdorado@mdoradoLaptop:~/pas/1718$ cat ejemploCarpeta.html
8 | <html>
9 | <head>
10 | <title>Listado del directorio ejemploCarpeta</title>
11 | </head>
12 | <body>
13 | <style type="text/css">
14 |   body { font-family: sans-serif;}
15 | </style>
16 | <h1>Listado del directorio ejemploCarpeta</h1>
17 | <ul>
18 | <li><strong>/home/mdorado/pas/1718/ejemploCarpeta/images</strong></li>

```

```

19 <ul><li><strong>/home/mdorado/pas/1718/ejemploCarpeta/images/Carpeta1</strong></li>
20 <ul><li>/home/mdorado/pas/1718/ejemploCarpeta/images/Carpeta1/elinfiernoexiste.jpg</li>
21 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Carpeta1/Linus_torvalds.jpg</li>
22 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Carpeta1/richard-stallman.jpg</li>
23 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Carpeta1/pirates.jpg</li>
24 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Carpeta1/devocion-administrador-de-sistemas.png</li>
25 </ul>
26 <li><strong>/home/mdorado/pas/1718/ejemploCarpeta/images/logos</strong></li>
27 <ul><li>/home/mdorado/pas/1718/ejemploCarpeta/images/logos/logo_eps_grande.jpg</li>
28 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/logos/Logo_uco.gif</li>
29 </ul>
30 <li><strong>/home/mdorado/pas/1718/ejemploCarpeta/images/Teoria</strong></li>
31 <ul><li>/home/mdorado/pas/1718/ejemploCarpeta/images/Teoria/a2.out</li>
32 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Teoria/HD.png</li>
33 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Teoria/centos.png</li>
34 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Teoria/debian.png</li>
35 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Teoria/Fedora.png</li>
36 <li>/home/mdorado/pas/1718/ejemploCarpeta/images/Teoria/raid0.png</li>
37 </ul>
38 </ul>
39 <li><strong>/home/mdorado/pas/1718/ejemploCarpeta/d1</strong></li>
40 <ul><li><strong>/home/mdorado/pas/1718/ejemploCarpeta/d1/d4</strong></li>
41 <ul><li>/home/mdorado/pas/1718/ejemploCarpeta/d1/d4/libsysfs.so.2</li>
42 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/d4/act_func.c</li>
43 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/d4/bcache.cpp</li>
44 </ul>
45 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/smo_routine.c</li>
46 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/compat.h</li>
47 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/ecoc.h</li>
48 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/svm_model_matlab.c</li>
49 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/libhandle.so.1.0.3</li>
50 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/audit.h</li>
51 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/auto_fs.h</li>
52 <li>/home/mdorado/pas/1718/ejemploCarpeta/d1/auditBackup.h</li>
53 </ul>
54 <li><strong>/home/mdorado/pas/1718/ejemploCarpeta/d2</strong></li>
55 <ul><li>/home/mdorado/pas/1718/ejemploCarpeta/d2/libglib-2.0.so.0</li>
56 <li>/home/mdorado/pas/1718/ejemploCarpeta/d2/a.out</li>
57 <li>/home/mdorado/pas/1718/ejemploCarpeta/d2/prueba.c</li>
58 <li>/home/mdorado/pas/1718/ejemploCarpeta/d2/ejemploSuid.sh</li>
59 </ul>
60 <li><strong>/home/mdorado/pas/1718/ejemploCarpeta/d3</strong></li>
61 <ul><li>/home/mdorado/pas/1718/ejemploCarpeta/d3/eje2</li>
62 <li>/home/mdorado/pas/1718/ejemploCarpeta/d3/disco2.png</li>
63 </ul>
64 <li>/home/mdorado/pas/1718/ejemploCarpeta/ejemploCarpeta.html</li>
65 </ul>
66 </body>
67 </html>

```

Al abrir el fichero en un navegador el resultado debería ser el que se muestra en la Figura 1.

Listado del directorio ejemploCarpeta/

- /home/mdorado/pas1718/ejemploCarpeta/ejemploCarpeta.html
- /home/mdorado/pas1718/ejemploCarpeta/d3
 - /home/mdorado/pas1718/ejemploCarpeta/d3/eje2
 - /home/mdorado/pas1718/ejemploCarpeta/d3/disco2.png
- /home/mdorado/pas1718/ejemploCarpeta/d1
 - /home/mdorado/pas1718/ejemploCarpeta/d1/audit.h
 - /home/mdorado/pas1718/ejemploCarpeta/d1/compat.h
 - /home/mdorado/pas1718/ejemploCarpeta/d1/svm_model_matlab.c
 - /home/mdorado/pas1718/ejemploCarpeta/d1/libhandle.so.1.0.3
 - /home/mdorado/pas1718/ejemploCarpeta/d1/auditBackup.h
 - /home/mdorado/pas1718/ejemploCarpeta/d1/smo_routine.c
 - /home/mdorado/pas1718/ejemploCarpeta/d1/d4
 - /home/mdorado/pas1718/ejemploCarpeta/d1/d4/bcache.cpp
 - /home/mdorado/pas1718/ejemploCarpeta/d1/d4/libsysfs.so.2
 - /home/mdorado/pas1718/ejemploCarpeta/d1/d4/act_func.c
 - /home/mdorado/pas1718/ejemploCarpeta/d1/auto_fs.hpp
 - /home/mdorado/pas1718/ejemploCarpeta/d1/auto_fs.h
 - /home/mdorado/pas1718/ejemploCarpeta/d1/ecoc.h
- /home/mdorado/pas1718/ejemploCarpeta/images
 - /home/mdorado/pas1718/ejemploCarpeta/images/Teoria
 - /home/mdorado/pas1718/ejemploCarpeta/images/Teoria/HD.png
 - /home/mdorado/pas1718/ejemploCarpeta/images/Teoria/centos.png
 - /home/mdorado/pas1718/ejemploCarpeta/images/Teoria/debian.png
 - /home/mdorado/pas1718/ejemploCarpeta/images/Teoria/raid0.png
 - /home/mdorado/pas1718/ejemploCarpeta/images/Teoria/a2.out
 - /home/mdorado/pas1718/ejemploCarpeta/images/Teoria/fedora.png
 - /home/mdorado/pas1718/ejemploCarpeta/images/Carpeta1
 - /home/mdorado/pas1718/ejemploCarpeta/images/Carpeta1/Linus_torvalds.jpg
 - /home/mdorado/pas1718/ejemploCarpeta/images/Carpeta1/devocion-administrador-de-sistemas.png
 - /home/mdorado/pas1718/ejemploCarpeta/images/Carpeta1/pirates.jpg
 - /home/mdorado/pas1718/ejemploCarpeta/images/Carpeta1/richard-stallman.jpg
 - /home/mdorado/pas1718/ejemploCarpeta/images/Carpeta1/elinfiernoexiste.jpg
 - /home/mdorado/pas1718/ejemploCarpeta/images/logos
 - /home/mdorado/pas1718/ejemploCarpeta/images/logos/Logo_uco.gif
 - /home/mdorado/pas1718/ejemploCarpeta/images/logos/logo_eps_grande.jpg
- /home/mdorado/pas1718/ejemploCarpeta/d2
 - /home/mdorado/pas1718/ejemploCarpeta/d2/ejemploSuid.sh
 - /home/mdorado/pas1718/ejemploCarpeta/d2/libglib-2.0.so.0
 - /home/mdorado/pas1718/ejemploCarpeta/d2/a.out
 - /home/mdorado/pas1718/ejemploCarpeta/d2/prueba.c

Figura 1: Fichero ejemploCarpeta.html de ejemplo para el ejercicio6.sh