

# Programación y Administración de Sistemas

## Práctica 1. Programación de *shell scripts*

Convocatoria de junio (curso 2014/2015)

Pedro Antonio Gutiérrez Peña

9 de marzo de 2015

### 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`. El día tope para la entrega de este guión de prácticas es el **domingo 15 de marzo a las 23.55h**. 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...`) y comprimir todos ellos en un fichero `.tar.gz` llamado `apellido1.apellido2.nombre.PASJun1415P1.tar.gz`.

## 1. `ejercicio1.sh`

Crear un *script* que reciba, como primer argumento, el nombre de una carpeta y, como segundo argumento (opcional), el nombre de un fichero con extensión `.tar.gz`. El *script* deberá comprobar que la carpeta existe y, en tal caso, comprimir todo su contenido dentro del fichero especificado. Solo deberán tenerse en cuenta aquellos ficheros cuya antigüedad sea mayor que 20 segundos. Para comprobar la antigüedad de un fichero puedes investigar sobre el comando `stat`. Si el segundo parámetro no se especifica deberemos utilizar como nombre de fichero `dir.tar.gz`, donde `dir` es el nombre del directorio. Un ejemplo de ejecución sería el siguiente:

```
1 i02gupep@NEWS:~/pas/1415$ ./ejercicio1.sh carpetaPrueba/
2 carpetaPrueba/dir2/libglib-2.0.so.0
3 carpetaPrueba/dir2/a.out
4 carpetaPrueba/dir2/prueba.c
5 carpetaPrueba/dir2/ejemploSuid.sh
6 carpetaPrueba/imagenes/Carpetal/Linus_torvalds.jpg
7 carpetaPrueba/imagenes/Carpetal/elinfiernoexiste.jpg
8 carpetaPrueba/imagenes/Carpetal/richard-stallman.jpg
9 carpetaPrueba/imagenes/Carpetal/devocion-administrador-de-sistemas.png
10 carpetaPrueba/imagenes/Carpetal/pirates.jpg
11 carpetaPrueba/imagenes/FotosOriginales/logo_eps_grande.jpg
12 carpetaPrueba/imagenes/FotosOriginales/Logo_uco.gif
13 carpetaPrueba/imagenes/Tema2/a2.out
14 carpetaPrueba/imagenes/Tema2/hd.png
15 carpetaPrueba/imagenes/Tema2/hdOrg.jpg
16 carpetaPrueba/imagenes/Tema2/raid0.png
17 carpetaPrueba/dir1/carpetaTercera/libsysfs.so.2
18 carpetaPrueba/dir1/carpetaTercera/act_func.c
19 carpetaPrueba/dir1/carpetaTercera/bcache.cpp
20 carpetaPrueba/dir1/smo_routine.c
21 carpetaPrueba/dir1/HMath.h
22 carpetaPrueba/dir1/compat_strncpy.h
23 carpetaPrueba/dir1/libtic.so.5
24 carpetaPrueba/dir1/adaboost_ecoc.h
25 carpetaPrueba/dir1/svm_model_matlab.c
```

```

26 carpetaPrueba/dir3/disco2
27 carpetaPrueba/dir3/eje2
28 Archivo carpetaPrueba.tar.gz creado correctamente
29 i02gupep@NEWS:~/pas/1415$ touch carpetaPrueba/fl
30 i02gupep@NEWS:~/pas/1415$ ./ejercicio1.sh carpetaPrueba/
31 carpetaPrueba/dir2/libglib-2.0.so.0
32 carpetaPrueba/dir2/a.out
33 carpetaPrueba/dir2/prueba.c
34 carpetaPrueba/dir2/ejemploSuid.sh
35 carpetaPrueba/imagenes/Carpeta1/Linus_torvalds.jpg
36 carpetaPrueba/imagenes/Carpeta1/elinfiernoexiste.jpg
37 carpetaPrueba/imagenes/Carpeta1/richard-stallman.jpg
38 carpetaPrueba/imagenes/Carpeta1/devocion-administrador-de-sistemas.png
39 carpetaPrueba/imagenes/Carpeta1/pirates.jpg
40 carpetaPrueba/imagenes/FotosOriginales/logo_eps_grande.jpg
41 carpetaPrueba/imagenes/FotosOriginales/Logo_uco.gif
42 carpetaPrueba/imagenes/Tema2/a2.out
43 carpetaPrueba/imagenes/Tema2/hd.png
44 carpetaPrueba/imagenes/Tema2/hdOrg.jpg
45 carpetaPrueba/imagenes/Tema2/raid0.png
46 carpetaPrueba/dir1/carpetaTercera/libsysfs.so.2
47 carpetaPrueba/dir1/carpetaTercera/act_func.c
48 carpetaPrueba/dir1/carpetaTercera/bcache.cpp
49 carpetaPrueba/dir1/smo_routine.c
50 carpetaPrueba/dir1/HMath.h
51 carpetaPrueba/dir1/compat_strncpy.h
52 carpetaPrueba/dir1/libtic.so.5
53 carpetaPrueba/dir1/adaboost_ecoc.h
54 carpetaPrueba/dir1/svm_model_matlab.c
55 carpetaPrueba/dir3/disco2
56 carpetaPrueba/dir3/eje2
57 Archivo carpetaPrueba.tar.gz creado correctamente
58 i02gupep@NEWS:~/pas/1415$ ./ejercicio1.sh carpetaPrueba/ # Transcurridos 20 segundos
59 carpetaPrueba/dir2/libglib-2.0.so.0
60 carpetaPrueba/dir2/a.out
61 carpetaPrueba/dir2/prueba.c
62 carpetaPrueba/dir2/ejemploSuid.sh
63 carpetaPrueba/imagenes/Carpeta1/Linus_torvalds.jpg
64 carpetaPrueba/imagenes/Carpeta1/elinfiernoexiste.jpg
65 carpetaPrueba/imagenes/Carpeta1/richard-stallman.jpg
66 carpetaPrueba/imagenes/Carpeta1/devocion-administrador-de-sistemas.png
67 carpetaPrueba/imagenes/Carpeta1/pirates.jpg
68 carpetaPrueba/imagenes/FotosOriginales/logo_eps_grande.jpg
69 carpetaPrueba/imagenes/FotosOriginales/Logo_uco.gif
70 carpetaPrueba/imagenes/Tema2/a2.out
71 carpetaPrueba/imagenes/Tema2/hd.png
72 carpetaPrueba/imagenes/Tema2/hdOrg.jpg
73 carpetaPrueba/imagenes/Tema2/raid0.png
74 carpetaPrueba/dir1/carpetaTercera/libsysfs.so.2
75 carpetaPrueba/dir1/carpetaTercera/act_func.c
76 carpetaPrueba/dir1/carpetaTercera/bcache.cpp
77 carpetaPrueba/dir1/smo_routine.c
78 carpetaPrueba/dir1/HMath.h
79 carpetaPrueba/dir1/compat_strncpy.h
80 carpetaPrueba/dir1/libtic.so.5
81 carpetaPrueba/dir1/adaboost_ecoc.h
82 carpetaPrueba/dir1/svm_model_matlab.c
83 carpetaPrueba/dir3/disco2
84 carpetaPrueba/dir3/eje2
85 carpetaPrueba/fl
86 Archivo carpetaPrueba.tar.gz creado correctamente

```

## 2. ejercicio2.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 (<ul></ul>) que estará compuesta de un conjunto de elementos (<ul></ul>). Para que las carpetas aparezcan en **negrita** puedes utilizar la etiqueta <strong></strong>. El nombre del fichero .html será dir.html, donde dir es el nombre de la carpeta. Un ejemplo de ejecución es el siguiente:

```

1 i02gupep@NEWS:~/pas/1415$ ./ejercicio2.sh .
2 Generando el listado de la carpeta /home/i02gupep/pas/1415 sobre el fichero 1415.html...
3 ¡Terminado!
4 i02gupep@NEWS:~/pas/1415$ ./ejercicio2.sh ./carpetaPrueba
5 Generando el listado de la carpeta /home/i02gupep/pas/1415/carpetaPrueba sobre el fichero
   carpetaPrueba.html...
6 ¡Terminado!
7 i02gupep@NEWS:~/pas/1415$ cat carpetaPrueba.html
8 <html>
9 <head>
10 <title>Listado del directorio ./carpetaPrueba/</title>
11 </head>
12 <body>
13 <style type="text/css">
14   body { font-family: sans-serif;}
15 </style>
16 <h1>Listado del directorio ./carpetaPrueba/</h1>
17 <ul>
18 <li><strong>./carpetaPrueba/dir2</strong></li>
19 <ul><li>./carpetaPrueba/dir2/libglib-2.0.so.0</li>
20 <li>./carpetaPrueba/dir2/a.out</li>
21 <li>./carpetaPrueba/dir2/prueba.c</li>
22 <li>./carpetaPrueba/dir2/ejemploSuid.sh</li>
23 </ul>
24 <li><strong>./carpetaPrueba/imagenes</strong></li>
25 <ul><li><strong>./carpetaPrueba/imagenes/Carpetal</strong></li>
26 <ul><li>./carpetaPrueba/imagenes/Carpetal/Linus_torvalds.jpg</li>
27 <li>./carpetaPrueba/imagenes/Carpetal/elinfiernoexiste.jpg</li>
28 <li>./carpetaPrueba/imagenes/Carpetal/richard-stallman.jpg</li>
29 <li>./carpetaPrueba/imagenes/Carpetal/devocion-administrador-de-sistemas.png</li>
30 <li>./carpetaPrueba/imagenes/Carpetal/pirates.jpg</li>
31 </ul>
32 <li><strong>./carpetaPrueba/imagenes/FotosOriginales</strong></li>
33 <ul><li>./carpetaPrueba/imagenes/FotosOriginales/logo_eps_grande.jpg</li>
34 <li>./carpetaPrueba/imagenes/FotosOriginales/Logo_uco.gif</li>
35 </ul>
36 <li><strong>./carpetaPrueba/imagenes/Tema2</strong></li>
37 <ul><li>./carpetaPrueba/imagenes/Tema2/a2.out</li>
38 <li>./carpetaPrueba/imagenes/Tema2/hd.png</li>
39 <li>./carpetaPrueba/imagenes/Tema2/hdOrg.jpg</li>
40 <li>./carpetaPrueba/imagenes/Tema2/raid0.png</li>
41 </ul>
42 </ul>
43 <li><strong>./carpetaPrueba/dir1</strong></li>
44 <ul><li><strong>./carpetaPrueba/dir1/carpetaTercera</strong></li>
45 <ul><li>./carpetaPrueba/dir1/carpetaTercera/libsysfs.so.2</li>
46 <li>./carpetaPrueba/dir1/carpetaTercera/act_func.c</li>
47 <li>./carpetaPrueba/dir1/carpetaTercera/bcache.cpp</li>
48 </ul>
49 <li>./carpetaPrueba/dir1/smo_routine.c</li>
50 <li>./carpetaPrueba/dir1/HMath.h</li>
51 <li>./carpetaPrueba/dir1/compat_strncpy.h</li>
52 <li>./carpetaPrueba/dir1/libtic.so.5</li>
53 <li>./carpetaPrueba/dir1/adaboost_ecoc.h</li>
54 <li>./carpetaPrueba/dir1/svm_model_matlab.c</li>
55 </ul>
56 <li><strong>./carpetaPrueba/dir3</strong></li>
57 <ul><li>./carpetaPrueba/dir3/disco2</li>
58 <li>./carpetaPrueba/dir3/eje2</li>
59 <li>./carpetaPrueba/dir3/disco</li>
60 <li>./carpetaPrueba/dir3/logo</li>
61 </ul>
62 </ul>
63 </body>
64 </html>

```

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

### 3. ejercicio3.sh

Escribir un *script* que realice las siguientes acciones<sup>1</sup>:

- Mostrar el número de archivos en el directorio actual, sin contar directorios.

<sup>1</sup>Puedes necesitar los comandos `wc` (para contar) y `tr` (para reemplazar caracteres)

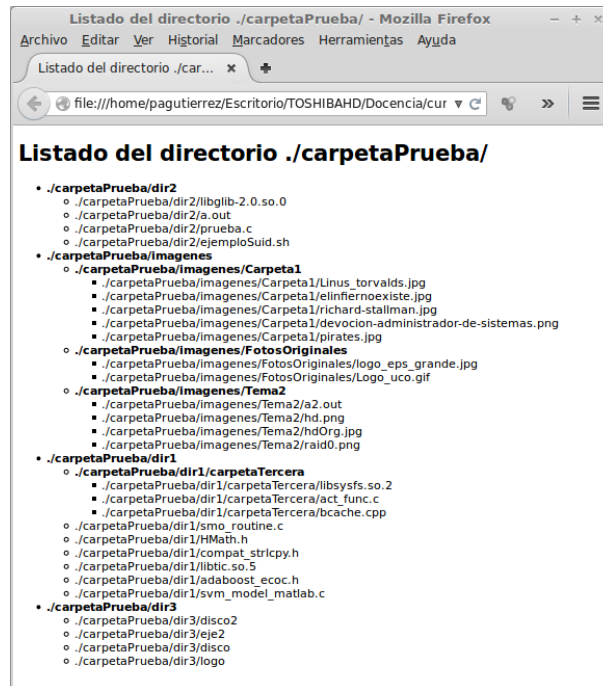


Figura 1: Fichero carpetaPrueba.html de ejemplo para el ejercicio2.sh

- Mostrar una lista ordenada de los usuarios que hay *logueados* en el sistema en el instante en que se ejecuta el *script*<sup>2</sup>.
- 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 i02gupep@NEWS:~/pas/1415$ cd carpetaPrueba
2 i02gupep@NEWS:~/pas/1415/carpetaPrueba$ ls
3 carpetaPrueba.html dir1 dir2 dir3 imagenes
4 i02gupep@NEWS:~/pas/1415/carpetaPrueba$ ../ejercicio3.sh
5 El número de archivos en la carpeta actual es 1
6 -----
7 Esta sería la lista de usuario conectados en este momento:
8 i02gupep
9 javier
10 malmacuf
11 p22loroj
12 tonin
13 jcheca
14 -----
15 ¿Qué carácter quieres contar? d
16 El carácter d aparece 18 veces en nombres de ficheros o carpetas contenidos en la carpeta actual
17 i02gupep@NEWS:~/pas/1415/carpetaPrueba$ cd dir1/
18 i02gupep@NEWS:~/pas/1415/carpetaPrueba/dir1$ ../../ejercicio3.sh
19 El número de archivos en la carpeta actual es 6
20 i02gupep
21 javier
22 malmacuf
23 p22loroj
24 tonin
25 jcheca
26 ¿Qué carácter quieres contar? T
27 El carácter T aparece 1 veces en nombres de ficheros o carpetas contenidos en la carpeta actual

```

<sup>2</sup>Consultar el comando who

## 4. ejercicio4.sh

Realizar un *script* que busque dentro de un fichero un determinado patrón, del mismo modo que hace `grep`. La diferencia es que este *script*, además de mostrar las líneas que coinciden con el patrón, deberá de mostrar una “ventana” alrededor de la línea. Esto es, se deberá imprimir la línea anterior a la emparejada, la línea emparejada y la línea posterior a la emparejada. Asegúrate de tratar los casos especiales de que la línea emparejada sea la primera o la última. Tienes que usar el comando `grep`, pero no puedes utilizar las opciones de contexto de `grep`: `-B`, `--before-context=NÚM`, `-A`, `--after-context=NÚM`, `-C`, `--context=NÚM`, `-NÚM`. El *script* recibe como primer argumento el nombre del fichero a analizar y como segundo argumento el patrón a buscar. Puede que necesites utilizar el comando `cut`.

A continuación se muestra un ejemplo de ejecución:

```
1 i02gupep@NEWTS:~/pas/1415$ cd carpetaPrueba
2 i02gupep@NEWTS:~/pas/1415/carpetaPrueba$ cd dir2
3 i02gupep@NEWTS:~/pas/1415/carpetaPrueba/dir2$ ../../ejercicio4.sh ejemploSuid.sh sudo
4 -----
5 Emparejamiento número: 1
6 gcc prueba.c
7 sudo mkdir directorio
8 sudo touch ./directorio/p1 ./directorio/p2
9 -----
10 Emparejamiento número: 2
11 sudo mkdir directorio
12 sudo touch ./directorio/p1 ./directorio/p2
13 sudo chown root a.out
14 -----
15 Emparejamiento número: 3
16 sudo touch ./directorio/p1 ./directorio/p2
17 sudo chown root a.out
18 sudo chgrp root a.out
19 -----
20 Emparejamiento número: 4
21 sudo chown root a.out
22 sudo chgrp root a.out
23 sudo chmod 701 a.out
24 -----
25 Emparejamiento número: 5
26 sudo chgrp root a.out
27 sudo chmod 701 a.out
28 ls -la
29 -----
30 Emparejamiento número: 6
31 set -x
32 sudo chmod u+s a.out
33 ls -la
34 -----
35 Emparejamiento número: 7
36 pause
37 sudo cat ./directorio/p1
38 i02gupep@NEWTS:~/pas/1415/carpetaPrueba/dir2$ ../../ejercicio4.sh ejemploSuid.sh /directorio/p.
39 -----
40 Emparejamiento número: 1
41 sudo mkdir directorio
42 sudo touch ./directorio/p1 ./directorio/p2
43 sudo chown root a.out
44 -----
45 Emparejamiento número: 2
46 echo "Ahora si te ha dejado!"
47 echo "Mostrando el contenido del fichero ./directorio/p1"
48 pause
49 -----
50 Emparejamiento número: 3
51 pause
52 sudo cat ./directorio/p1
```

## 5. ejercicio5.sh

Crear un *script* que reciba como argumento un parámetro que será un directorio y un segundo argumento que será una cadena. El *script* debe buscar todos los ficheros que estén alojados

en la carpeta que se pasa como primer argumento y cuyo nombre contenga en algún punto la cadena pasada como segundo argumento. Si no se especifica un segundo argumento, el *script* deberá mostrar todos los ficheros. Para cada fichero, habrá que mostrar<sup>3</sup>:

- La ruta completa que permite acceder al fichero.
- El tamaño del fichero.
- El número de referencias al i-nodo del fichero.
- La cadena de permisos.
- Un 1 si el fichero es ejecutable o un 0 si no lo es.

Todo estos campos se separarán por el carácter “;” y la salida deberá ordenarse numéricamente por tamaño de fichero (de menor a mayor). Un ejemplo de ejecución es el siguiente:

```

1 i02gupep@NEWS:~/pas/1415$ ./ejercicio5.sh carpetaPrueba/ jp
2 Ruta;Tamaño;ReferenciasInodo;Permisos;Ejecutable
3 carpetaPrueba/imagenes/Carpeta1/pirates.jpg;9836;1;-rw-----;0
4 carpetaPrueba/imagenes/Carpeta1/Linus_torvalds.jpg;17670;1;-rw-----;0
5 carpetaPrueba/imagenes/Carpeta1/richard-stallman.jpg;42999;1;-rw-----;0
6 carpetaPrueba/imagenes/FotosOriginales/logo_eps_grande.jpg;43161;1;-rw-----;0
7 carpetaPrueba/imagenes/Carpeta1/elinfiernoexiste.jpg;119847;1;-rw-----;0
8 carpetaPrueba/imagenes/Tema2/hdOrg.jpg;142700;1;-rw-----;0
9
10 i02gupep@NEWS:~/pas/1415$ ./ejercicio5.sh ./carpetaPrueba/
11 Ruta;Tamaño;ReferenciasInodo;Permisos;Ejecutable
12 ./carpetaPrueba/dir3/disco;22;1;lrwxrwxrwx;0
13 ./carpetaPrueba/dir3/logo;42;1;lrwxrwxrwx;0
14 ./carpetaPrueba/dir1/carpetaTercera/act_func.c;92;1;-rw-----;0
15 ./carpetaPrueba/dir1/svm_model_matlab.c;99;1;-rw-----;0
16 ./carpetaPrueba/dir1/smo_routine.c;104;1;-rw-----;0
17 ./carpetaPrueba/dir2/prueba.c;233;1;-rw-----;0
18 ./carpetaPrueba/dir1/compat_strncpy.h;370;1;-rw-----;0
19 ./carpetaPrueba/dir2/ejemploSuid.sh;483;1;-rw-----;0
20 ./carpetaPrueba/dir1/carpetaTercera/bcache.cpp;1233;1;-rw-----;0
21 ./carpetaPrueba/dir1/adaboost_ecoc.h;2377;1;-rw-----;0
22 ./carpetaPrueba/dir2/a.out;7275;1;-rwx-----;1
23 ./carpetaPrueba/dir3/eje2;7275;1;-rwx-----;1
24 ./carpetaPrueba/imagenes/Tema2/a2.out;7275;1;-rwx-----;1
25 ./carpetaPrueba/dir1/HMath.h;7991;1;-rw-----;0
26 ./carpetaPrueba/imagenes/Carpeta1/pirates.jpg;9836;1;-rw-----;0
27 ./carpetaPrueba/imagenes/Carpeta1/Linus_torvalds.jpg;17670;1;-rw-----;0
28 ./carpetaPrueba/imagenes/FotosOriginales/Logo_uco.gif;20005;1;-rw-----;0
29 ./carpetaPrueba/imagenes/Carpeta1/devocion-administrador-de-sistemas.png;21613;1;-rw-----;0
30 ./carpetaPrueba/dir1/carpetaTercera/libsysfs.so.2;38644;1;-rw-----;0
31 ./carpetaPrueba/imagenes/Carpeta1/richard-stallman.jpg;42999;1;-rw-----;0
32 ./carpetaPrueba/imagenes/FotosOriginales/logo_eps_grande.jpg;43161;1;-rw-----;0
33 ./carpetaPrueba/dir1/libtic.so.5;48572;1;-rw-----;0
34 ./carpetaPrueba/imagenes/Tema2/raid0.png;50683;1;-rw-----;0
35 ./carpetaPrueba/imagenes/Carpeta1/elinfiernoexiste.jpg;119847;1;-rw-----;0
36 ./carpetaPrueba/imagenes/Tema2/hdOrg.jpg;142700;1;-rw-----;0
37 ./carpetaPrueba/dir3/disco2;367419;1;-rw-----;0
38 ./carpetaPrueba/imagenes/Tema2/hd.png;367419;1;-rw-----;0
39 ./carpetaPrueba/dir2/libglib-2.0.so.0;822344;1;-rw-----;0

```

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

```

1 i02gupep@NEWS:~/pas/1415$ ./ejercicio5.sh carpetaPrueba/ > salida.csv
2 i02gupep@NEWS:~/pas/1415$ libreoffice4.0 salida.csv

```

## 6. ejercicio6.sh

Crear un *script* que “recolecte” ficheros conocidos en un conjunto de subcarpetas. El *script* recibirá un número de argumentos indeterminado, de manera que los ficheros se buscarán en

<sup>3</sup>Consulta las distintas opciones del comando `stat`

cada una de las carpetas que se pasen como argumento en la línea de comandos (sino se especifica ningún argumento, se tomará la carpeta actual). Para cada carpeta, se buscarán ficheros de los siguientes tipos:

- Ficheros ejecutables.
- Librerías: aquellos cuyo nombre empiece por “lib”.
- Código fuente: aquellos que tengan como extensión “.c” o “.cpp”.
- Ficheros de cabecera: aquellos que tengan como extensión “.h”.

El nombre de las carpetas a las que se copiarán los archivos encontrados se debe solicitar al usuario. Si el usuario no los escribe tras 5 segundos, se utilizarán los siguientes nombres:

- Ejecutables: “\$HOME/bin”.
- Librerías: “\$HOME/lib”.
- Código fuente: “\$HOME/src”.
- Ficheros de cabecera: “\$HOME/include”.

Si las carpetas existen, advertir al usuario y preguntar si sobrescribir ficheros, cuando sea necesario<sup>4</sup>. Todos los ficheros de código fuente encontrados se compilarán con el compilador correspondiente (gcc o g++) y los ejecutables generados (que tendrán el mismo nombre, pero sin extensión) se copiarán a la carpeta de ejecutables. Si alguna compilación falla, debe mostrarse un error por pantalla y en el fichero de log. Hacer un fichero de log con todas las operaciones de copiado, compilación o creación de carpetas que se vayan produciendo (./ejercicio6.log). El formato de este fichero de log es libre, pero debe contener fecha y hora para cada operación, origen y destino para las operaciones de copiado, y nombre de las carpetas creadas. Al finalizar la ejecución del *script* se debe mostrar:

- Número de directorios procesados.
- Número de ficheros copiados para cada una de las distintas categorías (si el fichero no se sobrescribió no debe contarse). Para los ficheros ejecutables, contar los compilados con éxito.
- Tiempo necesario para realizar todas las operaciones (en segundos).

Un ejemplo de ejecución es el siguiente:

```
1 i02gupep@NEWS:~/pas/1415$ ./ejercicio6.sh carpetaPrueba
2 Introduzca el directorio donde copiar los ejecutables:
3
4 Introduzca el directorio donde copiar las librerías:
5
6 Introduzca el directorio donde copiar los ficheros de código fuente:
7
8 Introduzca el directorio donde copiar las cabeceras:
9 Utilizando los ficheros:
10 /home/i02gupep/bin para almacenar los ficheros ejecutables
11 /home/i02gupep/lib para almacenar las librerías
12 /home/i02gupep/src para almacenar los ficheros de código fuente
13 /home/i02gupep/include para almacenar los ficheros de cabeceras
14 Procesando directorio carpetaPrueba...
15 Error al compilar el fichero carpetaPrueba/dir1/carpetaTercera/bcache.cpp
16 Número de directorios procesados: 1
17 Número de ficheros ejecutables: 7
18 Número de librerías: 3
19 Número de ficheros de código fuente: 5
20 Número de ficheros de cabecera: 3
21 Tiempo necesario: 3
```

<sup>4</sup>Es obligatorio hacer una función que dado un fichero origen y un fichero destino, compruebe si el destino existe, pregunte si el fichero se debe sobrescribir y opere en consecuencia.

```

22 i02gupep@NEWTS:~/pas/1415$ ls ~/bin ~/src ~/include ~/lib
23 /home/i02gupep/bin:
24 a2.out act_func a.out eje2 prueba smo_routine svm_model_matlab
25
26 /home/i02gupep/include:
27 adaboost_ecoc.h compat_strncpy.h HMath.h
28
29 /home/i02gupep/lib:
30 libglib-2.0.so.0 libsysfs.so.2 libtic.so.5
31
32 /home/i02gupep/src:
33 act_func.c bcache.cpp prueba.c smo_routine.c svm_model_matlab.c
34
35 i02gupep@NEWTS:~/pas/1415$ ./ejercicio6.sh carpetaPrueba/dir1/ carpetaPrueba/dir2/
36 Introduzca el directorio donde copiar los ejecutables: eee
37
38 Introduzca el directorio donde copiar las librerías: lll
39
40 Introduzca el directorio donde copiar los ficheros de código fuente:
41
42 Introduzca el directorio donde copiar las cabeceras:
43 Utilizando los ficheros:
44 eee para almacenar los ficheros ejecutables
45 lll para almacenar las librerías
46 /home/i02gupep/src para almacenar los ficheros de código fuente
47 /home/i02gupep/include para almacenar los ficheros de cabeceras
48 El directorio /home/i02gupep/src ya existía
49 El directorio /home/i02gupep/include ya existía
50 Procesando directorio carpetaPrueba/dir1/...
51 El fichero /home/i02gupep/src/bcache.cpp existe. ¿Desea sobrescribirlo? (S/N): S
52 Error al compilar el fichero carpetaPrueba/dir1/carpetaTercera/bcache.cpp
53 El fichero /home/i02gupep/src/act_func.c existe. ¿Desea sobrescribirlo? (S/N): N
54 El fichero /home/i02gupep/src/smo_routine.c existe. ¿Desea sobrescribirlo? (S/N): N
55 El fichero /home/i02gupep/src/svm_model_matlab.c existe. ¿Desea sobrescribirlo? (S/N): S
56 El fichero /home/i02gupep/include/HMath.h existe. ¿Desea sobrescribirlo? (S/N): S
57 El fichero /home/i02gupep/include/compat_strncpy.h existe. ¿Desea sobrescribirlo? (S/N): N
58 El fichero /home/i02gupep/include/adaboost_ecoc.h existe. ¿Desea sobrescribirlo? (S/N): N
59 Procesando directorio carpetaPrueba/dir2/...
60 El fichero /home/i02gupep/src/prueba.c existe. ¿Desea sobrescribirlo? (S/N): N
61 Número de directorios procesados: 2
62 Número de ficheros ejecutables: 5
63 Número de librerías: 3
64 Número de ficheros de código fuente: 2
65 Número de ficheros de cabecera: 1
66 Tiempo necesario: 27
67
68 i02gupep@NEWTS:~/pas/1415$ head ejercicio6.log
69 2015-02-22:11-25-12: Fichero carpetaPrueba/dir1/carpetaTercera/bcache.cpp copiado a /home/i02gupep/
70 src
71 2015-02-22:11-25-12: No se pudo compilar el fichero carpetaPrueba/dir1/carpetaTercera/bcache.cpp con
72 g++
73 2015-02-22:11-25-12: Fichero carpetaPrueba/dir2/prueba.c copiado a /home/i02gupep/src
74 2015-02-22:11-25-13: Fichero prueba copiado a /home/i02gupep/bin
75 2015-02-22:11-25-13: Fichero carpetaPrueba/dir2/prueba.c compilado con gcc
76 2015-02-22:11-25-13: Fichero carpetaPrueba/dir1/carpetaTercera/act_func.c copiado a /home/i02gupep/
77 src
78 2015-02-22:11-25-13: Fichero act_func copiado a /home/i02gupep/bin
79 2015-02-22:11-25-13: Fichero carpetaPrueba/dir1/carpetaTercera/act_func.c compilado con gcc
80 2015-02-22:11-25-13: Fichero carpetaPrueba/dir1/smo_routine.c copiado a /home/i02gupep/src
81 2015-02-22:11-25-13: Fichero smo_routine copiado a /home/i02gupep/bin

```