

# Piscina C Shell 00

Resumen: Este documento corresponde a la evaluación del módulo Shell 00 de la piscina de 42.

## Índice general

1.	Ilisti deciones	
II.	Preámbulo	3
III.	Ejercicio 00 : Z	4
IV.	Ejercicio 01 : testShell00	5
V.	Ejercicio 02 : Sí, quiero más	7
VI.	Ejercicio 03 : ¡Conéctame!	9
VII.	Ejercicio 04 : midLS	10
VIII.	Exercise 05 : GiT commit	11
IX.	Exercise 06 : gitignore	12
Χ.	Ejercicio 07 : diff	13
XI.	Ejercicio 08 : clean	14
XII.	Ejercicio 09 : Illusions, not tricks, Michael	15

#### Capítulo I

#### Instrucciones

- Esta página será la única referencia: no se fíe de los rumores de pasillo.
- Vuelva a leer bien los enunciados antes de entregar sus ejercicios. Los enunciados pueden cambiar en cualquier momento.
- Los ejercicios han sido ordenados con mucha precisión del más sencillo al más complejo. En ningún caso le prestaremos atención ni tendremos en cuenta un ejercicio complejo si no se ha conseguido realizar perfectamente un ejercicio más sencillo.
- Tenga cuidado con los permisos de sus archivos y de sus directorios.
- Debe respetar el procedimiento de entrega para todos sus ejercicios.
- Sus compañeros de piscina se encargarán de corregir sus ejercicios.
- Además de por sus compañeros, también será corregido por un programa que se llama la Moulinette.
- La Moulinette es muy estricta a la hora de dar notas. Está completamente automatizada. Es imposible discutir con ella sobre su nota. ¡Por lo tanto, sea riguroso!
- Los ejercicios shell se deben ejecutar con /bin/sh.
- <u>No debe</u> dejar en su directorio <u>ningún</u> archivo que no se haya indicado de forma explícita en los enunciados de los ejercicios.
- ¿Tiene alguna pregunta? Pregunte a su vecino de la derecha. Si no, pruebe con su vecino de la izquierda.
- Su manual de referencia se llama Google / man / Internet / ....
- ¡No olvide participar en el foro Piscina de su Intranet o en el slack de su Piscina!
- Lea detenidamente los ejemplos. Podrían exigir cosas que no se especifican necesariamente en los enunciados...

#### Capítulo II

#### Preámbulo

He aquí la letra de los créditos de Nicky Larson:

En la noche una sombra huye
Es un asesino que se escapa
Como un demonio sonríe
Su crimen quedará impune
Aparece un coche
Un frenazo, las ruedas crujen
Un disparo que resuena
La justicia se llama Nicky

[Estribillo]
En el calor
De la noche
El mal siempre será castigado
Ningún peligro le impresiona
Le atraen los golpes duros
Y la justicia le apasiona
Nicky Larson no le teme a nadie
Cuando los disparos retumban
Se mueve como un rayo
Sobre todo si la chica es guapa
Nicky Larson no le teme a nadie

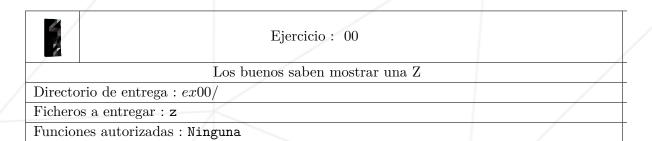
Como un cazador persigue a su presa Para que la justicia y el derecho Triunfen, está dispuesto a dar Su vida sin dudarlo Cuando aparece su silueta Los malos se ponen a temblar Saben que nunca podrán Escapar a este justiciero

#### [Estribillo]

Desgraciadamente, estos enunciados no tienen nada que ver con Nicky Larson.

## Capítulo III

Ejercicio 00 : Z

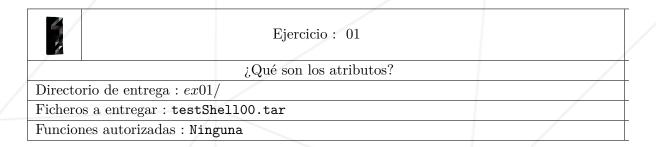


• Cree un archivo z que, cuando se ejecute un cat sobre él, muestre "Z" seguido de un salto de línea.

?>cat z
Z
?>

#### Capítulo IV

#### Ejercicio 01: testShell00



- Cree el archivo testShell00 en su directorio de entrega.
- Tiene que conseguir que:

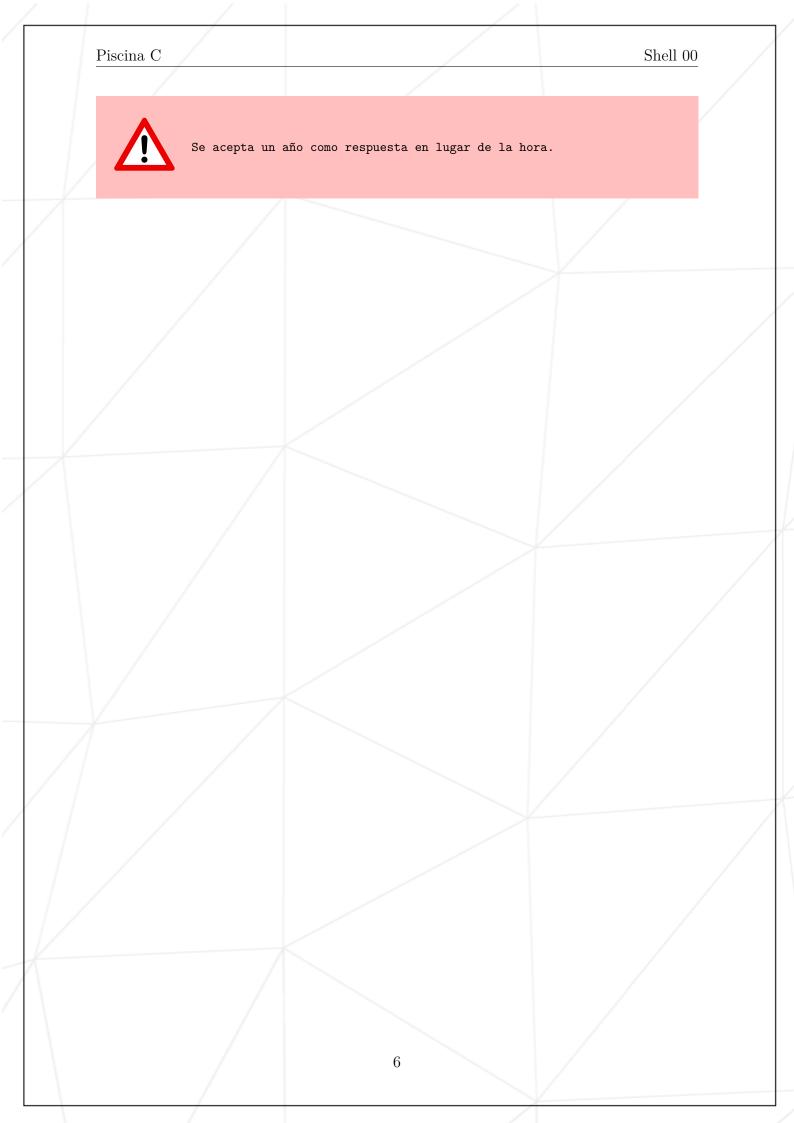
```
%> 1s -1
total XX
-r--r-xr-x 1 XX XX 40 Jun 1 23:42 testShell00
%>
```

Para las horas, se aceptará que se muestre el año si se ha superado la fecha del ejercicio (1 junio) en seis meses o más.

• Una vez resuelto el ejercicio, ejecute el comando tar -cf testShell00.tar testShell00 para crear el archivo que tiene que entregar.

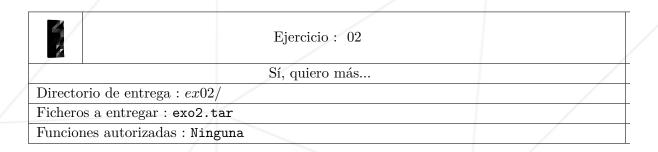


No se tendrán en cuenta los XX.



#### Capítulo V

#### Ejercicio 02 : Sí, quiero más...



• Cree todos estos archivos y directorios. Haga lo que sea necesario para que la ejecución de un ls -l en su directorio le muestre algo que se parezca a esto:

```
%> ls -1
total XX
drwx--xr-x 2 XX XX XX Jun 1 20:47 test0
-rwx--xr-- 1 XX XX 4 Jun 1 21:46 test1
dr-x--r-- 2 XX XX XX Jun 1 22:45 test2
-r---r-- 2 XX XX 1 Jun 1 23:44 test3
-rw-r---x 1 XX XX 2 Jun 1 23:43 test4
-r---r-- 2 XX XX 1 Jun 1 23:44 test5
lrwxr-xr-x 1 XX XX 5 Jun 1 22:20 test6 -> test0
%>
```

Para las horas, se aceptará que se muestre el año si se ha superado la fecha del ejercicio (1 junio) en seis meses o más.

• Una vez resuelto el ejercicio, ejecute el comando tar -cf exo2.tar \* para crear el archivo que tiene que entregar.

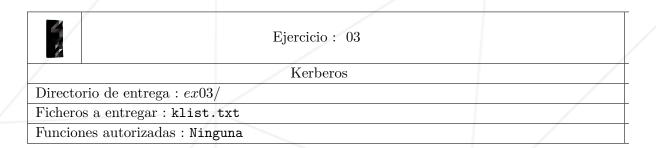


No se tendrán en cuenta los XX.

Piscina C		Shell 00
<b>^</b>		
Se acepta un año como respu	esta en lugar de la hora	
	8	

#### Capítulo VI

## Ejercicio 03: ¡Conéctame!



- Verifique que dispone de un tique Kerberos que no haya caducado u obtenga uno.
- Una vez que disponga de al menos un tique Kerberos, entregue la lista de todos sus tiques en un archivo klist.txt



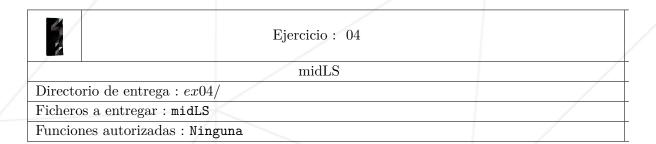
El nombre del archivo no se ha elegido al azar.



Estos comandos le van a resultar útiles durante toda la piscina, ino los olvide!

#### Capítulo VII

Ejercicio 04: midLS



• Escriba en un archivo midLS la línea de comandos que hay que ejecutar para listar los archivos y los directorios del directorio en curso, pero no los archivos ocultos, ni "."ni ".."(nada que empiece con un punto), separados por comas, ordenados por fecha de creación y de tal modo que los directorios vayan seguidos por el carácter slash ("/").



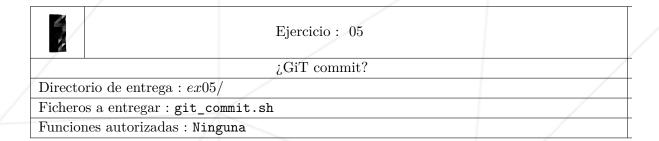
 $_{i}\,\text{No}$  hay que hacer lo que no se ha pedido!



RTFM

### Capítulo VIII

Exercise 05: GiT commit



• Cree un script shell qui devuelva los id de los últimos 5 commits de su repositorio git.

%> bash git\_commit.sh | cat -e
baa23b54f0adb7bf42623d6d0a6ed4587e11412a\$
2f52d74b1387fa80eea844969e8dc5483b531ac1\$
905f53d98656771334f53f59bb984fc29774701f\$
5ddc8474f4f15b3fcb72d08fcb333e19c3a27078\$
e94d0b448c03ec633f16d84d63beaef9ae7e7be8\$
%>

Su script será probado en nuestro entorno.



RTFM

#### Capítulo IX

Exercise 06: gitignore



• Escriba un script shell corto que devuelva la lista de archivos existentes ignorados por su repositorio git actualmente presentes en su repositorio local. Ejemplo:

```
%> bash git_ignore.sh | cat -e
.DS_Store$
mywork.c~$
%>
```

Su script será probado en nuestro entorno.



RTFM!

#### Capítulo X

Ejercicio 07: diff



Ejercicio: 07

Directorio de entrega : ex07/

Ficheros a entregar : b

Funciones autorizadas : Ninguna

• Cree el archivo b de tal modo que:

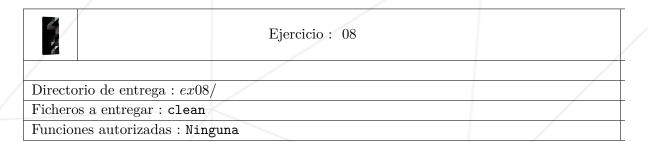
%>diff a b > sw.diff



man patch

#### Capítulo XI

#### Ejercicio 08 : clean



- Escriba en un archivo clean una línea de comandos que busque, a partir del directorio en curso y en todos sus subdirectorios, los archivos cuyo nombre se termine por ~, o empiece y se termine por #.
- La línea de comandos mostrará y borrará los archivos encontrados.
- Solo se autoriza un comando: nada de ';' , '&&' ni cosas por el estilo.



man find

#### Capítulo XII

Ejercicio 09 : Illusions, not tricks, Michael...

	Ejercicio: 09		
	Illusions, not tricks, Michael		
Directorio de entrega : $ex09/$			
Ficheros a entregar : ft_magic			
Funciones autorizadas : Ninguna			

• Cree un archivo mágico que se llame ft\_magic que tendrá que estar formateado correctamente para detectar los archivos de tipo 42 file y constituidos por la cadena "42" en el 42.º byte.



man file