



C Piscine

Shell 00

Resumen: Este documento corresponde al enunciado del módulo Shell 00 de la C Piscine de 42.

Versión: 4.6

Índice general

I.	Instrucciones	2
II.	Introducción	3
III.	Ejercicio 00: Z	5
IV.	Ejercicio 01: testShell00	6
V.	Ejercicio 02: Sí, quiero más...	8
VI.	Ejercicio 03: ¡Conéctame por SSH!	10
VII.	Ejercicio 04: midLS	11
VIII.	Exercise 05: GiT commit	12
IX.	Exercise 06: gitignore	14
X.	Ejercicio 07: diff	15
XI.	Ejercicio 08: clean	16
XII.	Ejercicio 09: Illusions, not tricks, Michael...	17
XIII.	Entrega y evaluación	18

Capítulo I

Instrucciones

- Esta página será la única referencia: no te fíes de los rumores.
- ¡Ten cuidado! Los enunciados pueden cambiar en cualquier momento.
- Los ejercicios han sido ordenados con mucha precisión, del más sencillo al más complejo. En ningún caso se tendrá en cuenta un ejercicio complejo si no se ha conseguido realizar perfectamente un ejercicio más sencillo.
- Asegúrate de que tus directorios y archivos tienen los permisos adecuados.
- Debes respetar el procedimiento de entrega para todos tus ejercicios.
- Tus compañeros de piscina se encargarán de corregir tus ejercicios.
- Además de por tus compañeros, también serán corregidos por un programa que se llama la Moulinette.
- La Moulinette es muy estricta a la hora de evaluar. Está completamente automatizada. Es imposible discutir con ella sobre tu nota. Por lo tanto, se extremadamente riguroso para evitar cualquier sorpresa.
- Los ejercicios shell se deben ejecutar con `/bin/sh`.
- No puedes dejar en tu directorio ningún archivo que no se haya indicado de forma explícita en los enunciados de los ejercicios.
- ¿Tienes alguna pregunta? Pregunta a tu compañero de la derecha. Si no, prueba con tu compañero de la izquierda.
- Tu manual de referencia se llama `Google / man / Internet / ...`
- Lee detenidamente los ejemplos. Podrían exigir cosas que no se especifican necesariamente en los enunciados...

Capítulo II

Introducción

Aquí tienes la letra de la canción “Moonlight Shadow”, tema principal de City Hunter:

The last time ever she saw him
Carried away by a moonlight shadow
He passed on worried and warning
Carried away by a moonlight shadow.
Lost in a riddle that Saturday night
Far away on the other side.
He was caught in the middle of a desperate fight
And she couldn't find how to push through

The trees that whisper in the evening
Carried away by a moonlight shadow
Sing a song of sorrow and grieving
Carried away by a moonlight shadow
All she saw was a silhouette of a gun
Far away on the other side.
He was shot six times by a man on the run
And she couldn't find how to push through

[Chorus]
I stay, I pray
See you in Heaven far away...
I stay, I pray
See you in Heaven one day.

Four A.M. in the morning
Carried away by a moonlight shadow
I watched your vision forming
Carried away by a moonlight shadow
A star was glowing in the silvery night
Far away on the other side
Will you come to talk to me this night
But she couldn't find how to push through


[Chorus]

Far away on the other side.
Caught in the middle of a hundred and five
The night was heavy and the air was alive
But she couldn't find how to push through
Carried away by a moonlight shadow
Carried away by a moonlight shadow
Far away on the other side.

Desgraciadamente, estos enunciados no tienen nada que ver con City Hunter.

Capítulo III

Ejercicio 00: Z

	Ejercicio: 00
Solo los mejores saben mostrar la Z	
Directorio de entrega: <i>ex00/</i>	
Archivos a entregar: z	
Funciones autorizadas: Ninguna	

- Crea un archivo **z** que, cuando se ejecute un **cat** sobre él, muestre “Z” seguido de un salto de línea.


```
?>cat z
Z
?>
```



Google es tu amigo

Capítulo IV

Ejercicio 01: testShell00

	Ejercicio: 01
¿Qué son los atributos de todas formas?	
Directorio de entrega: <i>ex01/</i>	
Archivos a entregar: testShell00.tar	
Funciones autorizadas: Ninguna	

- Crea el archivo **testShell00** en tu directorio de entrega.
- Encuentra la manera de que tu pantalla muestre lo siguiente (excepto por la línea de “total 1”):

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

- Una vez completados los pasos anteriores, ejecuta el siguiente comando para crear el archivo que debe ser entregado: **tar -cf testShell00.tar testShell00.**



No te preocupes por lo que aparece en lugar de "XX"




Se acepta un año como respuesta en lugar de la hora.



Le has preguntado a tu compi de la derecha?

Capítulo V

Ejercicio 02: Sí, quiero más...

	Ejercicio: 02
Sí, quiero más...	
Directorio de entrega: <i>ex02/</i>	
Archivos a entregar: exo2.tar	
Funciones autorizadas: Ninguna	

- Crea los siguientes archivos y directorios. Haz lo que sea necesario para que la ejecución de un `ls -l` en tu directorio muestre algo que se parezca a esto:

```
%> ls -l
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
lrwxrwxrwx 1 XX XX   5 Jun 1 22:20 test6 -> test0
%>
```

- Una vez resuelto el ejercicio, ejecuta el comando `tar -cf exo2.tar *` para crear el archivo que tendrás que entregar.



No te preocupes por lo que aparece en lugar de "XX"




Se acepta un año como respuesta en lugar de la hora.



No dudes en preguntarle a cualquier persona del cluster.

Capítulo VI

Ejercicio 03: ¡Conéctame por SSH!

	Ejercicio: 03
SSH Key	
Directorio de entrega: <i>ex03/</i>	
Archivos a entregar: <i>id_rsa_pub</i>	
Funciones autorizadas: Ninguna	

- Crea tu propia clave SSH. Cuando lo tengas:
 - Añade tu clave pública a tu repositorio, en un archivo *id_rsa_pub*
 - Actualiza tu clave ssh en la intranet. Esto te permitirá lanzar el repositorio a nuestro servidor git.



El nombre del archivo no ha sido elegido de manera aleatoria.



Asegúrate de entender la diferencia entre clave pública y clave privada.



Le has preguntado a tu compi de la izquierda?

Capítulo VII

Ejercicio 04: midLS

	Ejercicio: 04
midLS	
Directorio de entrega: <i>ex04/</i>	
Archivos a entregar: midLS	
Funciones autorizadas: Ninguna	

- Crea en un archivo **midLS** que contenga el comando que hay que ejecutar para listar los archivos y los directorios del directorio actual, pero no los archivos ocultos, ni `."`ni `."`(nada que empiece con un punto), separados por una coma y un espacio, ordenados por fecha de modificación y de tal modo que los directorios vayan seguidos por el carácter slash (`/`).



¡No hay que hacer lo que no se ha pedido!




RTFM!



Git push con frecuencia

Capítulo VIII

Exercise 05: GiT commit

	Ejercicio: 05
¿GiT commit?	
Directorio de entrega: <i>ex05/</i>	
Archivos a entregar: <i>git_commit.sh</i>	
Funciones autorizadas: Ninguna	

- Crea un script de shell que muestre en pantalla los id de los últimos 5 commits de tu repositorio git.

```
%> bash git_commit.sh | cat -e
baa23b54f0adb7bf42623d6d0a6ed4587e11412a$
2f52d74b1387fa80eea844969e8dc5483b531ac1$
905f53d98656771334f53f59bb984fc29774701f$
5ddc8474f4f15b3fcb72d08fcb333e19c3a27078$
e94d0b448c03ec633f16d84d63beaef9ae7e7be8$
%>
```

Para probar tu script, utilizaremos nuestro propio entorno.



RTFM!



El primer retry es corto, no dudes en pedir una evaluación intermedia para medir tu avance!


Milestone conseguido, sigue así!

Has alcanzado el mínimo de ejercicios para validar este proyecto. Tu decides si quieres continuar con el resto de ejercicios opcionales o saltar al siguiente proyecto. Ambos caminos te serán útiles el día de mañana. Para tomar una decisión informada, considera las siguientes razones:

- El primer examen es sobre programación en C. Puede que quieras probar con el primer proyecto de C antes de hacerlo. Lo mismo para el primer rush hacia el final de esta semana (ya aprenderás lo que es un rush).
- Tu excelencia en esta piscina será evaluada en base a múltiples factores. Completar cada proyecto es uno de ellos, pero el progreso a lo largo de toda la lista de proyectos de la piscina es otro. Elige sabiamente para optimizar tus resultados.
- Siempre podrás intentar el mismo proyecto de nuevo en un par de días/semanas, hasta el final de la piscina.
- Mantente sincronizado con tus compañeros para facilita una mejor colaboración.

Capítulo IX

Exercise 06: gitignore

	Ejercicio: 06
GiT	
Directorio de entrega: <i>ex06/</i>	
Archivos a entregar: <code>git_ignore.sh</code>	
Funciones autorizadas: Ninguna	

- Crea un script de shell que muestre por pantalla la lista de archivos existentes ignorados por tu repositorio git **actualmente presentes** en tu repositorio local. Ejemplo:

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

Para probar tu script, utilizaremos nuestro propio entorno.




RTFM!



Déja que los demás te inspiren, no que te hagan el trabajo

Capítulo X

Ejercicio 07: diff

	Ejercicio: 07
Directorio de entrega: <i>ex07/</i>	
Archivos a entregar: <i>b</i>	
Funciones autorizadas: Ninguna	

- Crea el archivo *b* de tal modo que:

```
%>cat -e a
STARWARSS$
Episode IV, A NEW HOPE It is a period of civil war.$
$
Rebel spaceships, striking from a hidden base, have won their first victory against the evil
Galactic Empire.$
During the battle, Rebel spies managed to steal secret plans to the Empire's ultimate weapon, the
DEATH STAR,$
an armored space station with enough power to destroy an entire planet.$
$
Pursued by the Empire's sinister agents, Princess Leia races home aboard her starship, custodian of
the stolen plans that can save her people and restore freedom to the galaxy...$
$

%>diff a b > sw.diff
```




man patch



No te fíes de ninguna fuente de información: Haz siempre tus propios test, controles y verificaciones.

Capítulo XI

Ejercicio 08: clean

	Ejercicio: 08
Directorio de entrega: <i>ex08/</i>	
Archivos a entregar: clean	
Funciones autorizadas: Ninguna	

- Escriba en un archivo **clean** una línea de comandos que busque, a partir del directorio actual 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 otros trucos.




`man find`



La colaboración es la clave del éxito

Capítulo XII

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

	Ejercicio: 09
Ilusiones, no trucos, Michael...	
Directorio de entrega: <i>ex09/</i>	
Archivos a entregar: ft_magic	
Funciones autorizadas: Ninguna	

- Crea 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 byte 42°.



`man file`



Fallar es parte de tu aprendizaje.

Capítulo XIII

Entrega y evaluación

Entrega tu proyecto en tu repositorio `Git` como de costumbre. Solo el trabajo entregado en el repositorio será evaluado durante la defensa. No dudes en comprobar varias veces los nombres de los archivos para verificar que sean correctos.



Sólo necesitas entregar los archivos requeridos por el enunciado de este proyecto.