

Práctico 0: ShellScripting

Sistemas Operativos - 2025

Según Wikipedia, el Shell es *el programa informático que provee una interfaz de usuario para acceder a los servicios del sistema operativo*. Comenzaremos la materia familiarizándonos con esta herramienta, que usarán el resto de sus carreras, y con los comandos más comunes que pueden acceder.

Este práctico consiste de una serie de ejercicios que se resuelven con **una** línea de comando ejecutada en la consola. Necesitarán descargar los archivos de datos dados en sus computadoras para comprobar si el comando que proponen funciona correctamente.


Archivos con datos necesarios:

- atpplayers.in - Lista de jugadores de tenis profesionales
- superliga.in - Tabla de posiciones de la Superliga
- weather_cordoba.in - Datos meteorológicos de Córdoba
- heroes.csv - Datos con la lista completa de superhéroes

Recursos de ayuda:

- Bite Size Linux.pdf - Guía rápida de comandos Linux
- explainshell.com - Explicación detallada de comandos

Tendrán que **investigar por su cuenta** cómo funcionan los distintos comandos (como `grep`, `cat`, `sort`, `head`, `awk`, `tr`, `cut`, `wc`), conectores de comandos como `&` ; pipes `|` y redirecciones `><`.

 **Nota importante sobre IA:** Si usan ChatGPT para copiar la consigna y luego pegar la respuesta, entonces no aprenden nada y sólo están perdiendo su tiempo (y el nuestro). Si quieren usar algún asistente de IA, les recomendamos usar un prompt que

explícite cosas como: "estoy intentando aprender este tema, dame alguna pista pero no me des la respuesta completa".

Ejercicios

Ejercicio 1: Información del procesador

¿Qué modelo de procesador tiene tu equipo? Cuando necesitamos un dato del equipo, como por ejemplo el modelo del procesador, podemos buscar en el archivo `/proc/cpuinfo`. Sin embargo, este comando nos devuelve un listado muy largo y tienen que encontrar la forma de buscar sólo la información necesaria.

Comandos útiles: `cat`, `grep`, `head` **Objetivo:** Mostrar sólo el modelo del procesador (una línea)

Ejercicio 2: Número de cores

Si la computadora tiene más de una unidad de ejecución (multi-core) seguramente en el punto anterior se repitió más de una vez el modelo del procesador. Averiguar cómo usar `wc` para poder determinar cuántas unidades de ejecución tiene el procesador, aprovechando los comandos utilizados en el ejercicio 1.

Comandos útiles: `cat`, `grep`, `wc` **Objetivo:** Mostrar sólo el número de cores (un número)

Ejercicio 3: Red social de superhéroes

Usted tiene la maravillosa idea de crear una red social para superhéroes donde puedan interactuar sin máscaras y con sus nombres verdaderos (todo financiado por Bruce Wayne o Tony Stark, dependiendo del estudio). Para la semana de lanzamiento, le enviará una invitación personalizada a cada superhéroe. Su base de datos es el siguiente archivo [heroes.csv](#).

Notar que: La segunda columna contiene el nombre real de cada superhéroe, pero los usuarios en su sistema deben estar todos en minúsculas y sin espacios (deben separar los nombres y apellidos con el caracter "_" .). ¿Cómo obtendría esta lista de nombres de usuarios con un único comando bash? Incluso la descarga del archivo. Puntos extra si eliminan las líneas en blanco del resultado. Los que no tienen identidad(Identity) no incluirlos en el listado final.

Comandos útiles: `curl`, `cut`, `tr`, `grep`, `sed` **Objetivo:** Crear un archivo `superheroes_usuarios.txt` con los nombres procesados. Ejemplo del contenido del archivo final:

```
richard_milhouse_jones  
abraxas  
emil_blonsky  
sharon_carter  
gabriel_lan
```


Ejercicio 4: Análisis meteorológico

Parte A: Día de máxima temperatura

El archivo `weather_cordoba.in` contiene mediciones meteorológicas realizadas en un día en Córdoba. Las primeras 3 columnas corresponden al año, mes y día de las mediciones. Las restantes 6 columnas son la temperatura media, la máxima, la mínima, la presión atmosférica, la humedad y las precipitaciones medidas de ese día.

Objetivo: Calcular el día de máxima temperatura (máximo de máximas).

Idea para facilitar la tarea: Ordenar los días según su temperatura máxima. Puntos extra por mostrar en la pantalla sólo la fecha de dicho día.

 **Importante:** Para evitar los números reales, los grados están expresados en décimas de grados (por ejemplo, 15.2 grados está representado por 152 décimas). La presión también ha sido multiplicada por 10 y las precipitaciones por 100, o sea que están expresadas en centésimas de milímetro.

Comandos útiles: `sort`, `head`, `awk` **Objetivo:** Mostrar la fecha del día más caluroso

Parte B: Día de mínima temperatura

Objetivo: Calcular el día de mínima temperatura (mínimo de mínimas).

Idea para facilitar la tarea: Ordenar los días según su temperatura mínima. Puntos extra por mostrar en la pantalla sólo la fecha de dicho día.

Comandos útiles: `sort`, `head`, `awk` **Objetivo:** Mostrar la fecha del día más frío

Ejercicio 5: Ranking de tenistas

El archivo `atpplayers.in` es un listado por orden alfabético de jugadores profesionales de tenis. El nombre del jugador viene acompañado de una abreviatura de su país, el número que ocupa en el ranking, su edad, su puntaje y el número de torneos jugados en el último año.

Objetivo: Ordenar el listado de jugadores según la posición en el ranking.

Comandos útiles: `sort` **Nota:** El ranking está en la tercera columna

Ejercicio 6: Tabla de Superliga

El archivo `superliga.in` contiene datos con el siguiente formato: nombre de equipo sin espacios, puntos, partidos jugados, partidos ganados, partidos empatados, partidos perdidos, goles a favor y goles en contra.

Objetivo: Ordenar la tabla del campeonato de la Superliga según la cantidad de puntos, y desempatar por diferencia de goles.

Comandos útiles: `sort`, `awk` **Nota:** La diferencia de goles se calcula como (goles a favor - goles en contra)

Ejercicio 7: MAC address

¿Cómo ver la MAC address de nuestro equipo? Investiguen el comando `ip`. En el manual de `grep` van a encontrar la especificación de muchas operaciones, por ejemplo `-o`, `-i`, y

muchas más. Algo muy utilizado son las expresiones regulares para realizar una búsqueda. En el manual de `grep` tienen un apartado donde explica su uso. Con esta información deberían poder construir una secuencia de comandos de shell para imprimir por consola la MAC address de su interfaz ethernet.

Comandos útiles: `ip`, `grep`, expresiones regulares **Objetivo:** Mostrar solo la MAC address de la interfaz ethernet

Ejercicio 8: Renombrar archivos de subtítulos

Supongamos que bajaron una serie de televisión completa con subtítulos (de forma completamente legal, por supuesto). Sin embargo, los subtítulos tienen el sufijo `_es` en el nombre de cada archivo y para que puedan ser reproducidos en el televisor, que nunca fue demasiado smart, el archivo de subtítulos tiene que tener exactamente el mismo nombre que el archivo de video con la extensión `srt`. La serie tiene más de 100 capítulos, es imposible realizar los cambios uno a uno.

Parte 1: Para emular la situación, crear una carpeta con el nombre `serie_prueba` y dentro de ella crear 10 archivos con nombres que sigan el formato `fma_S01EXX_es.srt`, donde XX es el número de capítulo, desde 1 hasta 10.

Parte 2: Con un segundo comando (el que usarían realmente), cambiar el nombre de cada archivo sacando el sufijo `_es`.

Comandos útiles: `mkdir`, `touch`, `for`, `mv` **Ayuda:** Usar un ciclo `for`. Se pueden hacer ciclos en una única línea. Puntos extra si los archivos de video no siguen un patrón, sino que pueden llamarse de cualquier forma.

Ejercicio 9: Edición multimedia con FFmpeg [OPCIONAL]

El comando `ffmpeg` sirve para editar streams de video y audio desde la consola, de forma muchísimo más rápida que otros editores de video. Tienen que descubrir cómo realizar las siguientes tareas:

Parte A: Recortar video

Recortar un video. Pueden usar SimpleScreenRecorder (linux) para grabar un pequeño video de prueba. Luego, usen ffmpeg para sacarle los primeros y los últimos segundos en donde se ve el la pantalla como inician y cortan la grabación.

Comandos útiles: `ffmpeg` **Objetivo:** Crear un video recortado eliminando los primeros y últimos segundos

Parte B: Mezclar audio

Mezclar streams de audio. Graben un pequeño audio de voz, y descarguen de internet alguna pista de sonido que quieran poner de fondo. Usen `ffmpeg` para superponer las dos pistas. Tomen como ejemplo los podcasts de Nico!

Comandos útiles: `ffmpeg` **Objetivo:** Crear un archivo de audio mezclando voz y música de fondo

Nota: Este ejercicio es opcional pero recomendado para aprender herramientas multimedia

Formato de Entrega - IMPORTANTE

Para que el evaluador automático pueda procesar correctamente su entrega, **DEBEN** seguir estrictamente el siguiente formato:

Estructura del archivo .sh

Su archivo de entrega debe tener la siguiente estructura exacta:

```
None
#!/bin/bash

# EJERCICIO 1: Modelo del procesador
[su comando aquí]
```

EJERCICIO 2: Número de unidades de ejecución (cores)

[su comando aquí]

EJERCICIO 3: Lista de nombres de usuarios para red social de superhéroes

[su comando aquí]

EJERCICIO 4A: Día de máxima temperatura en Córdoba

[su comando aquí]

EJERCICIO 4B: Día de mínima temperatura en Córdoba

[su comando aquí]

EJERCICIO 5: Ordenar jugadores de tenis por ranking

[su comando aquí]

EJERCICIO 6: Ordenar tabla de Superliga por puntos y diferencia de goles

[su comando aquí]

EJERCICIO 7: MAC address de la placa WiFi del equipo

[su comando aquí]

EJERCICIO 8: Renombrar archivos de subtítulos

[su comando aquí]

EJERCICIO 9A: Recortar video con ffmpeg

[su comando aquí]

EJERCICIO 9B: Mezclar audio con ffmpeg

[su comando aquí]

Reglas estrictas que deben seguir:

1. **Un comando por ejercicio:** Cada ejercicio debe resolverse con exactamente una línea de comando.
2. **Comentarios obligatorios:** Cada comando debe estar precedido por un comentario que indique el número de ejercicio y una breve descripción.
3. **Orden secuencial:** Los ejercicios deben estar en orden del 1 al 9, sin saltar ninguno.
4. **Ejercicio 4 dividido:** El ejercicio 4 debe entregarse como dos comandos separados: 4A (máxima temperatura) y 4B (mínima temperatura).
5. **Ejercicio 9 dividido:** El ejercicio 9 debe entregarse como dos comandos separados: 9A (recortar video) y 9B (mezclar audio).
6. **Sin líneas vacías:** No dejen líneas vacías entre comandos.
7. **Sin comandos comentados:** No comenten los comandos con `#` al inicio de la línea del comando.
8. **Archivo ejecutable:** Asegúrense de que el archivo tenga permisos de ejecución.

Ejemplo de entrega correcta:

None

```
#!/bin/bash
```

```
# EJERCICIO 1: Modelo del procesador
```

```
[su comando aquí]
```

```
# EJERCICIO 2: Número de unidades de ejecución (cores)
```

```
[su comando aquí]
```


EJERCICIO 3: Lista de nombres de usuarios para red social de superhéroes

[su comando aquí]

EJERCICIO 4A: Día de máxima temperatura en Córdoba

[su comando aquí]

EJERCICIO 4B: Día de mínima temperatura en Córdoba

[su comando aquí]

EJERCICIO 5: Ordenar jugadores de tenis por ranking

[su comando aquí]

EJERCICIO 6: Ordenar tabla de Superliga por puntos y diferencia de goles

[su comando aquí]

EJERCICIO 7: MAC address de la placa WiFi del equipo

[su comando aquí]

EJERCICIO 8: Renombrar archivos de subtítulos

[su comando aquí]

EJERCICIO 9A: Recortar video con ffmpeg

[su comando aquí]

EJERCICIO 9B: Mezclar audio con ffmpeg

[su comando aquí]

Errores comunes que evitar:

✗ NO hacer esto:

```
None
# Comando comentado (no se ejecutará)
# [su comando aquí]

# Múltiples comandos en una línea
[comando1] && [comando2]
[comando1] ; [comando2]

# Líneas vacías entre comandos
[su comando aquí]

[su comando aquí]

# Sin comentarios descriptivos
[su comando aquí]
```

✓ Sí hacer esto:

```
None
# EJERCICIO 1: Modelo del procesador
[su comando aquí]
# EJERCICIO 2: Número de unidades de ejecución (cores)
[su comando aquí]
```

Nomenclatura del archivo de entrega:

IMPORTANTE: El nombre del archivo debe seguir este formato exacto:

None

apellido_nombre_dni.sh

Ejemplos correctos:

- `garcia_maria_12345678.sh`
- `lopez_carlos_87654321.sh`
- `rodriguez_ana_11223344.sh`
- `garcialopez_mariajose_12345678`

Ejemplos incorrectos:

- `entrega.sh` ❌ (sin identificación)
- `practico0.sh` ❌ (sin identificación)
- `garcia maria 12345678.sh` ❌ (espacios en lugar de guiones bajos)
- `GARCIA_MARIA_12345678.sh` ❌ (mayúsculas)
- `garcia_lopez_maria_jose_12345678.sh` ❌ (separación entre los apellidos y los nombres)

Reglas para el nombre del archivo:

1. **Apellido** en minúsculas. Si tienen más de un apellido pueden poner sólo el primero o todos juntos sin separadores.
2. **Guión bajo** como separador
3. **Nombre** en minúsculas. Si tienen más de un apellido pueden poner sólo el primero o todos juntos sin separadores.
4. **Guión bajo** como separador
5. **DNI** sin puntos ni espacios
6. **Extensión** `.sh`

Verificación antes de entregar:

- Asegúrense de que su archivo tenga exactamente 11 comandos (ejercicios 1-9, con 4A, 4B, 9A, 9B separados).

- Verifiquen que cada comando funcione correctamente en su sistema.
- Comprueben que los archivos de datos estén en la carpeta **datos/** relativa al directorio donde ejecutan los comandos.
- Para el ejercicio 3, verifiquen que se cree el archivo **superheroes_usuarios.txt**.
- Para el ejercicio 8, asegúrense de que exista la carpeta **serie_prueba/** con archivos de prueba con el sufijo **_es.srt**

```
$ ls serie_prueba
fma_S01E01_es.srt fma_S01E03_es.srt fma_S01E05_es.srt fma_S01E07_es.srt fma_S01E09_es.srt
fma_S01E02_es.srt fma_S01E04_es.srt fma_S01E06_es.srt fma_S01E08_es.srt fma_S01E10_es.srt
```

- **Verifiquen que el nombre del archivo siga el formato:**

apellido_nombre_dni.sh

⚠ Nota importante: Si no siguen este formato exacto, el evaluador automático no podrá procesar su entrega correctamente y recibirán 0 puntos.

Entrega

La resolución de este Laboratorio es individual. Deben entregar un archivo .sh a través del en el siguiente formulario <https://forms.gle/bur8yFBFUXx4HQGZ7> siguiendo estrictamente el formato especificado en la sección anterior. El archivo debe contener exactamente 11 comandos (uno por cada ejercicio, con los ejercicios 4 y 9 divididos en subpartes, si no completan algún ejercicio o el opcional deben escribir el siguiente comando en la respuesta **echo no lo hice**) y debe nombrarse según el formato **apellido_nombre_dni.sh**.

Criterios de evaluación:

- **Formato correcto:** Estructura del archivo según las especificaciones
- **Funcionalidad:** Cada comando debe ejecutarse correctamente
- **Nomenclatura:** Nombre del archivo con formato requerido
- **Compleitud:** Todos los ejercicios no opcionales deben estar resueltos

El práctico no se evalúa con nota, pero debe estar entregado y aprobado para poder promocionar o regularizar la materia.

La entrega debe ser enviada antes del día 19/08/2025 a las 15:59h