

SISTEMAS OPERATIVOS

TRABAJO PRÁCTICO 1:

"Primeros pasos en LINUX"

Objetivos del práctico

Al terminar este trabajo Ud. habrá aprendido a:

1. Utilizar convenientemente algunos de los principales comandos del SO LINUX.
2. Configurar dispositivos de almacenamiento secundario (disketteras/imágenes)
3. Conocer editores y familiarizarse con el que mejor se adapte a su requerimiento.
4. Generar archivos ejecutables (archivos interpretados-scripts).

Herramientas necesarias:

Para resolver los ejercicios propuestos necesitará:

1. Una PC con SO XP/VISTA/W7 con el emulador VMWARE.
2. El material (copia DVD) de la Cátedra.

Fuentes de Información sugeridas

Encontrará información útil en:

- Páginas de manual de LINUX
- El material provisto por la Cátedra.
- Guía de clases de laboratorio: Uso de Comandos Básicos de Linux - El Shell -
- <http://www.debian.org/doc/manuals/reference/ch-tutorial.es.html>
- <http://www.tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html>
- Kernighan, Brian W.; Pike, Rob. (1984). The Unix Programming Enviroment, Prentice Hall.
- Tutorial BASH de MacProgramadores
- Newham, Cameron. (2005). Learning the Bash Shell, Third Edition, O'Reilly Media
- Shotts, William E., Jr - The Linux Command Line <http://linuxcommand.org/tlcl.php>
- <http://tldp.org/LDP/abs/html/tests.html>

Requisitos de Entrega

Lugar y Fecha de entrega:

- 1 La fecha de entrega para este práctico será informada por el CAMPUS en el momento de publicar el TP.
- 2 Los trabajos deben ser entregados vía e-mail a la dirección de correo: sistemasoperativosutnsantafe@gmail.com en el asunto deberá indicar: "TP1 - GRUPO XX" (XX es el número que identifica al grupo).
- 3 No se aceptarán trabajos incompletos.

Formato de Entrega.

Deberá enviar dos archivos con la resolución del trabajo:

1. La imagen de un diskette en formato ext2 conteniendo los scripts.
2. El segundo, es un archivo de texto. Deberá reunir las siguientes características:

1. Secciones del documento (Todas obligatorias):
 - 1.1. **Carátula de presentación:** Debe incluir OBLIGATORIAMENTE:
 - 1.1.1. Asignatura
 - 1.1.2. Número y Descripción del trabajo práctico
 - 1.1.3. Año y Cuatrimestre de Cursado
 - 1.1.4. Identificación del Grupo
 - 1.1.5. Nombres, Apellidos y direcciones de correo electrónico de TODOS los Integrantes del grupo
 - 1.2. **Sección Principal:** Aquí debe incluirse la resolución de cada uno de los problemas planteados. Para cada respuesta debe indicarse OBLIGATORIAMENTE, el número y título del problema al que corresponde tal como aparece en el enunciado.
 - 1.3. **Sección de Descargos:** Aquí debe incluirse cualquier comentario que deba tenerse en cuenta para la corrección del práctico. Use esta sección para indicar cosas como:
 - Qué no pudo resolver alguno de los problemas
 - Qué no pudo resolver COMPLETAMENTE alguno de los problemas.
 - Qué no está seguro si el problema está resuelto correctamente.

Comentar los problemas en esta sección es la única forma de obtener puntaje parcial para un ítem que no está bien resuelto. Si se encuentra un problema no resuelto o resuelto de manera INCOMPLETA y eso no está comentado en esta sección, perderá puntos adicionales (no sólo le descontaremos puntos por el error sino también por no avisarnos). Si no tiene ningún comentario, deje esta sección en blanco.

Penalizaciones.

Los prácticos entregados en fechas posteriores al límite fijado, tendrán una quita de puntos. Para ver el método empleado para restar puntos consulte en el Campus.

Cambios al enunciado del práctico, fechas de entrega, etc.

Cualquier cambio en los enunciados, fechas de entrega, etc. será informado utilizando dos métodos:

1. El campus virtual.
2. La lista de correos.

El alumno no puede alegar que no estaba al tanto de los cambios si esos cambios fueron anunciados utilizando alguno de los dos métodos.

SUGERENCIA: Consulte frecuentemente las novedades del Curso en el Campus Virtual y asegúrese de que ha sido incorporado a la lista de correos.

Honestidad académica:

Está bien hablar entre los grupos acerca de cómo resolver problemas, pero los grupos son de hasta 3 integrantes.

No entregue el trabajo de otras personas como propio. Tampoco entregue trabajos publicados en Internet como propios sin citar las fuentes.

Cualquier trabajo, porción de trabajo o texto sin la cita correspondiente es plagio.

Cada grupo debe mantener su código para sí mismo, si su proyecto es copiado, puede ser difícil determinar quién es el verdadero autor.

Cualquier ayuda que reciba deberá documentarla como un comentario al inicio del programa. Por ejemplo, si encuentra una solución a un ejercicio en un texto o manual, debería citar la fuente. Una razonable ayuda, no afectará la aprobación de los trabajos pero fallas al citar las fuentes o la ausencia de las mismas es fraude.

Queda debidamente aclarado, que los trabajos son de autoría, desarrollo y elaboración propia y no de un tercero.

Por último, la Cátedra podrá solicitar revisar el Trabajo Práctico con coloquio presencial con los alumnos del grupo en caso de considerarlo.

COMANDOS BÁSICOS

EJERCICIO 1.

Crear un script llamado **tiempoVivido.sh** que solicitará por teclado como ingreso, una fecha de nacimiento y calcule la edad, los días vividos, las horas vividas, los minutos y segundos vividos.

Ejemplos de corrida:

```
so2011# ./tiempoVivido.sh
```

Salida:

Ingrese la fecha de nacimiento (dd-mm-aaaa): 10-09-1975

Edad = 45

Días Vividos = 16425

Horas Vividas = 394200

Minutos Vividos = 23652000

Segundos Vividos = 1419120000

so2011#

NOTA: Es requisito resolver este ejercicio mediante el comando **date**, además de otros comandos necesarios para realizar los cálculos. Tenga en cuenta o asuma para simplificar, que el año tiene 365 días. No se consideran bisiestos o cualquier otra complejidad adicional.

GESTIÓN DE SISTEMA DE ARCHIVOS y DIRECTORIOS

EJERCICIO 2.

Escribir un script llamado **testArch.sh** que distinga si el primer argumento es un archivo o un directorio. Si es un archivo, deberá imprimir de qué tipo de archivo se trata, y si el argumento es un directorio deberá mostrar el contenido del mismo.

Al finalizar no importando el caso de si es archivo o directorio deberá mostrar el tamaño.

Si la ruta pasada como parámetro no existe, deberá mostrar un mensaje NO EXISTENTE!

Ejemplo de ejecución:

Caso Directorio

```
so2011:~/2020/TP1/scriptsTP# ./testArch.sh /root/  
/root/ Es un directorio.
```

```
2013 2016 2019 cosbastion.txt makefile slackware.txt TP2011
```

```
2014 2017 2020 DEADJOE molclock.c SOFLOPPY.TXT
```

```
2015 2018 a.out Desktop out.txt src
```

```
Tamaño1: 4096
```

```
Tamaño2: 171M /root/
```

```
so2011:~/2020/TP1/scriptsTP#
```

Caso Archivo

```
so2011:~/2020/TP1/scriptsTP# ./testArch.sh /root/cosbastion.txt
/root/cosbastion.txt: ASCII text
Tamaño1: 38896
Tamaño2: 40K /root/cosbastion.txt
so2011:~/2020/TP1/scriptsTP#
```

Caso Inexistente

```
so2011:~/2020/TP1/scriptsTP# ./testArch.sh /root/coVid19
/root/coVid19 NO EXISTENTE!
so2011:~/2020/TP1/scriptsTP#
```

MANIPULACIÓN DE VARIABLES

EJERCICIO 3.

a) Generar el script **entorno.sh** que deberá mostrar el siguiente mensaje:

"Bienvenido usuario: xxxx "

usted se ha conectado en el día: xx/yy/zzzz (4 caracteres para el año)
a la hora: xxhs:yym:zzs
en la terminal: xxx del host yyy

Donde en los valores "xxx", "" en cada caso se deberá mostrar las salidas correspondientes obtenidas de las variables de entorno adecuadas o mediante parseo de comandos adecuados.

```
so2011:~/2019/TP1/original# ./entorno.sh
Bienvenido usuario: root
Usted se ha conectado en el día: 29/08/2019
A la hora: 08h:02m:18s
En la terminal: xterm del host: so2011
Se encuentran conectados en este momento, los siguientes usuarios:
root      pts/0      2019-08-29 08:01 (192.168.1.1)
Su path es:
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
so2011:~/2019/TP1/original#
```

b) Crear un script **crear_env.sh** que cree una nueva variable de entorno llamada **nuevoEntorno** en base a un parámetro recibido y muestre su valor.

Ejemplo de corrida

```
so2011# ./crear_env.sh 1
```

Variable "nuevoEntorno" creada
Valor de la variable 1

EJERCICIO 4.

Crear un script **bits.sh** que solicite un número entero **p** por teclado de tal manera que ese número sea la potencia en base 2, es decir 2^p (**2** elevado a **p**) y que se pueda expresar en:

Bits,
Bytes,
Kilobytes,
Megabytes,
Gigabytes,
Terabytes y
Petabytes

Ejemplo de corrida:

```
so2011:~/2014/TP1# ./bits.sh
Potencia de 2 ->
62
```

```
-----
4611686018427387904 bits
576460752303423488 bytes
562949953421312 kilobytes
549755813888 megabytes
536870912 gigabytes
524288 terabytes
512 petabytes
```

GESTIÓN DE PROCESOS

EJERCICIO 5.

Cree el script **procesos.sh** que busque todos los archivos recursivamente desde el **directorio raíz** que contengan un punto "." en su nombre y los almacene en un archivo XXX que recibirá como parámetro.

Mientras se realiza la búsqueda se deberá liberar el shell para enviar el siguiente mensaje **"Se está generando el listado de archivos que contienen un punto"**.

Antes y después del mensaje se deberá mostrar por pantalla el listado de procesos que se encuentran activos.

Enuncie brevemente sus conclusiones.

REDIRECCIONES

EJERCICIO 6.

Escriba un script que se llame **redirecciones.sh** que recibirá como argumento el nombre de un directorio y una "a" o una "z".

Si es a, mostrará el contenido del directorio ordenado ascendentemente en forma alfabética, si es una z lo mostrara en forma descendente.

Además si recibe una f almacenara el resultado en un archivo llamado **orden.txt**

Ejemplo de ejecución:

```
root@node0:/home/SO2015# ./redirecciones.sh /bin z
znew
zmore
zless
...
...
bzcat
busybox
bunzip2
bash
root@node0:/home/SO2015# ./redirecciones.sh /bin z f
root@node0:/home/SO2015# ls
ayuda.sh procesos.sh status.sh tiempo.sh
orden.txt redirecciones.sh testArch.sh
root@node0:/home/SO2015# cat orden.txt
znew
zmore
zless
zgrep
...
...
bzip2recover
bzip2
bzgrep
bzfgrep
bzexe
bzegrep
bzdifff
bzcmmpp
bzcat
busybox
bunzip2
bash
```