

Ejercicio 1:

Utilizaremos el archivo de contribuyentes de la AFIP, el cual puede descargar del siguiente vínculo:
<http://www.afip.gob.ar/genericos/clincripcion/archivos/apellidoNombreDenominacion.zip>

Y tiene la explicación sobre formato y definición de los campos en la página de descarga:
<http://www.afip.gob.ar/genericos/clincripcion/archivoCompleto.asp>

Ejecute el siguiente script y responda las preguntas que se detallan a continuación del mismo, tenga en cuenta que antes de poder ejecutarlo deberá otorgarle permisos de ejecución.

FORMATO DE ENTREGA: Se deberá entregar el script en un archivo con extensión .sh y las respuestas contenidas en el mismo como líneas comentadas.

```
#!/bin/bash

if [ $# != 2 ]; then
    echo "..." exit
fi

if ! [ -f "$2" ]; then
    echo "..."
    exit
fi

X=$1

awk 'BEGIN {
    FIELDWIDTHS = "11 30"
}
$1 == "$X" {
    print "Nombre"
    print "-----"
    print $2 " $1"
    exit
}' $2
```

Responda:

- ¿Qué significa la línea "#!/bin/bash"?
- ¿Con qué comando y de qué manera otorgó los permisos de ejecución al script?
- Explique el objetivo general del script.
- Complete los echo "..." con mensajes acordes a cada situación según lo que se valida y renombre la variable X con un nombre acorde a su utilidad.
- Explique de manera general la utilidad de AWK.
- Es posible ejecutar este script para procesar varios archivos a la vez en una única llamada? Ejemplifique.

IMPORTANTE: No copiar el archivo de padrón en la entrega del ejercicio.

Ejercicio 2:

Mejore el script del punto anterior, para que se pueda filtrar tanto por CUIT, DNI o por nombre o parte del mismo, dependiendo del modificador -c, -d o -n respectivamente.
También mejore la salida para que muestre un detalle de toda la información disponible.
Y al final un resumen de cantidad de tipos de contribuyente filtrados.

IMPORTANTE: No copiar el archivo de padrón en la entrega del ejercicio.

Criterios de corrección:

Control	Criticidad
El script ofrece ayuda con -h, -? o -help explicando cómo se lo debe invocar	Obligatorio
Valida cantidad de parámetros mínimos.	Obligatorio
Funciona correctamente según enunciado.	Obligatorio
Resolver las 3 formas de filtro en una única llamada de AWK.	Deseable
Funciona con rutas relativas, absolutas o con espacios.	Obligatorio

Ejercicio 3:

Una agencia de lotería quiere controlar los cartones de bingo, Los mismos se encuentran cargados en un archivo de texto, el cual recibirá por parámetro. En cada línea del archivo contiene el número de cartón y 15 números separados por Tabs, cada 5 números representan una línea del cartón. El script al recibir la señal SIGUSR1 debe generar y mostrar por pantalla un número aleatorio del 0 al 99 (no debe repetir el número sorteado, ya que hay sólo una bolilla por número) y realizar el control de los cartones para mostrar cuándo un cartón obtuvo línea o bingo, mostrando también por pantalla los resultados. Al finalizar mostrará por pantalla un resumen de cantidad de bolillas sorteadas y los números de cartón ganadores de línea y bingo.

Debe omitir el CTRL + C y continuar ejecutando hasta encontrar el ganador.

Nota: Tal cual las reglas del bingo solo el primer cartón que complete línea será el ganador por línea y cuando se "cante" bingo el juego finaliza.

Cartones.txt

NumeroCarton	n1	n2	n3	n4	n5	n6	n7	n8	n9	n10	n11	n12	n13	n14	n15
1234	5	9	15	47	98	66	72	61	0	12	3	25	34	57	1
9999	10	78	45	75	0	26	47	68	81	3	12	55	69	1	33
1272	29	88	69	22	55	31	5	51	90	27	20	33	0	14	8

Criterios de corrección:

Control	Criticidad
El script ofrece ayuda con -h, -? o -help.	Obligatorio
Valida parámetros correctamente.	Obligatorio
Funciona correctamente según enunciado.	Obligatorio
Se adjuntan archivos de prueba por parte del grupo.	Obligatorio
Funciona con rutas relativas, absolutas o con espacios.	Obligatorio
Se implementan funciones	Opcional

Ejercicio 4:

Se desea realizar un script para controlar la similitud de un archivo base contra los archivos contenidos en un directorio. El script comparará el archivo pasado por parámetro contra cada uno de los archivos encontrados en una ruta de directorios también pasado por parámetro, informando los archivos que tienen cierto porcentaje de igualdad. Si no se informa el directorio, tomará el directorio actual.

El script recibirá por parámetro el archivo base, la ruta de directorio (opcional) y un porcentaje mínimo de igualdad.

Criterios de corrección:

Control	Criticidad
El script ofrece ayuda con -h, -? o -help.	Obligatorio
Valida parámetros.	Obligatorio
Funciona correctamente según enunciado.	Obligatorio
Se adjuntan archivos de prueba por parte del grupo.	Obligatorio
Funciona con rutas relativas, absolutas o con espacios.	Obligatorio
Se implementan funciones	Deseable

Ejercicio 5:

Realizar un script que genere un backup comprimido de una ruta especificada.

El script recibirá los siguientes parámetros:

- Ruta de origen.
- -t para backupear todos los archivos o
- -x [extensión] para backupear los archivos con dicha extensión.

El nombre del archivo backup debe contener la fecha y hora de generación (al segundo) y en caso de ser por extensión, debe figurar la misma en el nombre.

Debe mantener los últimos 5 backups de cada tipo, eliminando el resto automáticamente y generar un archivo de log, con el mismo nombre del backup informando usuario de generación, fecha y hora y el listado de archivos resguardados para permitir una revisión rápida de que backup contiene cierto archivo sin tener que analizar los archivos comprimidos.

Criterios de corrección:

Control	Criticidad
El script ofrece ayuda con -h, -? o -help.	Obligatorio
Valida parámetros.	Obligatorio
Funciona correctamente según enunciado.	Obligatorio
Se adjuntan archivos de prueba por parte del grupo.	Obligatorio
Funciona con rutas relativas, absolutas o con espacios.	Obligatorio
Se implementan funciones	Deseable

Ejercicio 6:

Realizar un script que renombre masivamente los archivos de un directorio.

El mismo deberá tomar un patrón de búsqueda en el nombre de los archivos y un valor de reemplazo. El script recibirá por parámetro el directorio a analizar, el patrón a buscar y el texto de reemplazo.

DirectorioA:

Pepexx.log

Pedroxx.log

Pacoxx.log

Llamada al script:

reemplazar.sh DirectorioA xx 20190328

Resultado:

Pepe20190328.log

Pedro20190328.log

Paco20190328.log

Criterios de corrección:

Control	Criticidad
El script ofrece ayuda con -h, -? o -help.	Obligatorio
Valida parámetros.	Obligatorio
Funciona correctamente según enunciado.	Obligatorio
Se adjuntan archivos de prueba por parte del grupo.	Obligatorio
Funciona con rutas relativas, absolutas o con espacios.	Obligatorio
Se implementan funciones	Deseable