

**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)**

Formato para prácticas de laboratorio

PROGRAMA EDUCATIVO	PLAN DE ESTUDIO	CLAVE DE UNIDAD DE APRENDIZAJE	NOMBRE DE LA UNIDAD DE APRENDIZAJE
Ingeniero en Computación	2009-2	12124	Taller de Sistema Operativo Unix

PRÁCTICA No.	LABORATORIO DE	Taller de Sistema Operativo Unix	DURACIÓN (HORAS)
15	NOMBRE DE LA PRÁCTICA	Expresiones regulares	2

1. INTRODUCCIÓN

Una expresión regular es una cadena de caracteres que puede también contener símbolos especiales llamados metacaracteres, esto son caracteres especiales o comodines como `. * ? + [] () { } ^ $ | y \`. Las expresiones regulares se utilizan junto con algunos mandos para realizar búsquedas avanzadas dentro de archivos de texto. Por eso se puede decir que las expresiones regulares no hacen únicamente la descripción formal de un patrón de texto sino que además son un buscador de cadenas de texto.

2. OBJETIVO (COMPETENCIA)

Crear expresiones regulares utilizando metacaracteres y el mando `grep` para realizar búsquedas dentro de archivos de texto con actitud analítica y positiva.

3. FUNDAMENTO

La notación que se presenta a continuación permite escribir patrones para secuencias de caracteres (cadenas), y corresponde a la que con ligeras variantes se usa en diversas utilidades para proceso de textos con: "`grep`", "`awk`", "`perl`" y otros mandos de Linux.

Formuló Eva Herrera Ramirez, G. Angelica Flores Rivera	Revisó Aglay González Pacheco	Aprobó	Autorizó David I. Rosas Almeida
Nombre y Firma del Maestro	Nombre y Firma del Responsable de Programa Educativo	Nombre y Firma del Responsable de Gestión de Calidad	Nombre y Firma del Director / Representante de la Dirección

Expresiones simples

Son patrones que se ajustan a un único símbolo (en general, a un único carácter):

Símbolo	Significado
x	Carácter x, si es carácter normal
.	Cualquier carácter
[aeiou]	Un carácter del conjunto
[a-z]	Un carácter del rango
[^aeiou]	Cualquier carácter que no sea vocal minúscula
\x	Carácter x, si x es un carácter especial (metacaracter)
^	Principio del texto, va al principio de la expresión
\$	Fin del texto, va al final de la expresión
\<	Principio de la palabra
\>	Fin de la palabra

A continuación se muestran algunos ejemplos utilizando estos símbolos con expresiones regulares y el mando *grep* que se vio en la práctica anterior, para ello utilizaremos el archivo *simple.txt*.

```
eherrera@computacion:~$ cat simple.txt
Casa    caso    cazar   azar
Casado  calzado pala   besar
beso    vaso    jara    taza
jala    paso    lazo    rezar.
```

Como se puede observar nos muestra la línea en la cual esta 'v' como patrón.

```
eherrera@computacion:~$ grep 'v' simple.txt
beso    vaso    jara    taza
```

En este ejemplo nos muestra las líneas en las cuales se encuentre un patrón que inicie con cualquier carácter (representado por el `.`) seguido por `aso`.

```
eherrera@computacion:~$ grep '.aso' simple.txt
Casa    caso    cazar   azar
beso    vaso    jara    taza
jala    paso    lazo    rezar.
```

En este caso nos muestra las líneas cuyo patrón tenga los caracteres **ja** seguidos de una **l** o una **r** y que continúen con **a**.

```
eherrera@computacion:~$ grep 'ja[lr]a' simple.txt
beso   vaso   jara   taza
jala   paso   lazo   rezar.
```

Con esta expresión se muestra la línea que tenga un punto como carácter, como el punto es un metacaracter se debe anteponer a este la diagonal inversa “\” (slash).

```
eherrera@computacion:~$ grep '\.' simple.txt
jala   paso   lazo   rezar.
```

En este ejemplo nos muestra las líneas que inicien con **b**.

```
eherrera@computacion:~$ grep '^b' simple.txt
beso   vaso   jara   taza
```

Nos muestra las líneas que terminan con **o**.

```
eherrera@computacion:~$ grep 'o$' simple.txt
beso   vaso   jara   taza
```

Esta expresión muestra las líneas en la cuales hay palabras que inicien con **pa**.

```
eherrera@computacion:~$ grep '\<pa' simple.txt
Casado calzado pala   besar
jala   paso   lazo   rezar.
```

Ahora nos muestra las líneas de las palabras que terminen con **so**.

```
eherrera@computacion:~$ grep 'so\>' simple.txt
Casa   caso   cazar   azar
beso   vaso   jara   taza
jala   paso   lazo   rezar.
```

Expresiones compuestas o extendidas

Algunos mandos, como egrep o grep -E, aceptan expresiones regulares extendidas, que comprenden las expresiones regulares simples más algunos operadores que permiten construcciones más complejas. Los operadores incorporados son los siguientes:

Caracter	Significado
?	El carácter anterior es opcional y se buscará como máximo una vez.

Caracter	Significado
*	El carácter anterior se buscará cero o más veces.
+	El carácter anterior se buscará una o más veces.
{ }	Establecen el número de veces que se repetirá el carácter que lo precede en la expresión.
{n}	El carácter anterior se buscará exactamente n veces. (aparece n o más veces)
{n,}	El carácter anterior se buscará n o más veces, para $n \neq 2$
{,m}	El carácter anterior se buscará como máximo m veces, para $m \neq 2$
{n,m}	El carácter anterior se buscará al menos n veces, pero no más de m veces.(no funciona)

Ahora buscaremos patrones dentro del archivo `arch.txt` haciendo uso de la simbología anterior y expresiones regulares en combinación con el mando **egrep -E**, **egrep** hace la misma acción.

```
eherrera@computacion:~$ cat arch.txt
abc
babd
abbc
abbcd
abbbcd
axbcde
cccdea
```

La expresión contiene las letras **a,b** y **c** donde **b** aparece una o más veces

```
eherrera@computacion:~$ grep -E 'ab+c' arch.txt
abc
abbc
abbcd
abbbcd
```

La expresión contiene las letras **a,b** y **c** donde **b** puede aparecer cero o más veces

```
eherrera@computacion:~$ grep -E 'ab*c' arch.txt
abc
abbc
abbcd
abbbcd
```

La expresión contiene las letras **a** y **c**, es opcional que **b** aparezca

```
eherrera@computacion:~$ grep -E 'ab?c' arch.txt
abc
```

Muestra las líneas que contengan la secuencia **ab** o **de**.

```
eherrera@computacion:~$ grep -E 'ab|de' arch.txt
abc
babd
abbc
abbc d
abbbcd
axbcde
cccdea
```

\$ grep -E 'ab{2,3}' arch.txt La expresión contiene **ab** juntas y puede tener de dos a tres repeticiones de **b**

```
eherrera@computacion:~$ grep -E 'ab{2,3}' arch.txt
abbc
abbc d
abbbcd
```

4. PROCEDIMIENTO (DESCRIPCIÓN)

A) EQUIPO NECESARIO

Computadoras con Linux instalado

MATERIAL DE APOYO

Práctica impresa

B) DESARROLLO DE LA PRÁCTICA

1. Iniciar la sesión de Linux
2. Crear un archivo al cual llamaremos Palabras.txt
**Ayudar a saturar la alfarda lastrar taladrar susurra
saludar agarrar gutural la alfaguara yugular a *halagar
garatusa Frugal atalaya a trastada fraguar al guardar
gayata al asaltar yarda una aguja lasaña la falda
arrabal manta valvula umbral tarumba brutal trasvasar murmurar
llamarada ralladura barba alambra rambla salvar mandar turnar
¿Para que utilizar expresiones regulares?
untar aguantar agravar bramar navaja gafar naval haba
dual vaya guarda ablandar barra marrana gamma ganga.**
3. Mostrar las líneas que inician con vocal.
4. Mostrar las líneas que terminen con consonantes.
5. Mostrar las líneas que tengan como patrón un asterisco
6. Mostrar las líneas que tengan palabras de 3 caracteres.
7. Mostrar las líneas con palabras que inicien con a y que terminen con r.
8. Mostrar las líneas que tengan el patrón val, tal o ral.
9. Mostrar las líneas que no tengan el patrón e.

10. Mostrar las líneas que tengan signos de interrogación.
11. Mostrar las líneas que tengan 2 o mas l's o 2 o mas r's consecutivas.
12. Utilizando grep, tuberías y expresiones regulares, muestre en pantalla solo los directorios del directorio */usr*.
13. Muestre los usuarios del archivo *passwd* cuyo nombre inicia con *po* donde la *o* aparece dos veces que se encuentra en */etc/*
14. Muestre los archivos de texto de extensión *.txt* que se encuentran en */externos/home/clases/compartidos/*.
15. Mostrar los nombres de los directorios de su directorio home.
16. Mostrar únicamente los archivos ejecutables de su directorio home.
17. Muestre únicamente los usuarios cuyo nombre inicia con *po* o con *tu* del archivo *passwd* que se encuentra en */etc/*
18. Muestre todos los archivos de *etc* que contienen *twain.example.com* o *bronto.pangaea.edu* como nombre de host.

C) CÁLCULOS (SI APLICA) Y REPORTE

El alumno será evaluado por su profesor ya sea mediante un examen escrito o por medio de la entrega de un reporte.

5. RESULTADOS Y CONCLUSIONES

El alumno será capaz de crear expresiones regulares simples y extendidas utilizando metacaracteres.

6. ANEXOS

7. REFERENCIAS

<http://manpages.ubuntu.com/manpages/hardy/man1/grep.1.html#contenttoc4>