Expressões Regulares

regex

Fábio dos Reis

Expressões Regulares

Expressões Regulares são conjuntos de caracteres usados para procurar e manipular textos em arquivos, pastas, etc, baseados em padrões.

Podem ser usadas com inúmeros comandos do Linux e em linguagens de programação.

Metacaracteres

Caracteres que possuem significado especial.
Transformam caracteres literais em expressões poderosas.
São os seguintes:
\ . * + - { } [ ] ^ $ | ? ( ) : ! =

Os metacaracteres podem ter mais de um significado, dependendo do contexto de uso.
Obs. Aspas NÃO são metacaracteres,

Metacaracteres

Caracter único

Representamos qualquer caracter único com um ponto .

grep ".orte" arquivo #Retorna as linhas que contém a string orte, independente da primeira letra.

grep -i "f.b" /etc/passwd

Metacaracteres - escape

E se quiséssemos usar o ponto . como caractere literal, e não metacaractere?
Então usaremos outro metacaractere para indicar o "escape" do ponto: a barra \
A barra dará ao caractere na sequência um significado alternativo - em vez de metacaractere, será tratado como caractere literal normal.
Ex.: Imagine um texto contendo as strings 2.000, 2,000 e 2-000. Teste as regex a seguir e veja a diferença:
grep 2.00 string
grep 2\.00 string

Conjuntos ou Classes de Caracteres

- São listas de caracteres escritas dentro de colchetes [ ] e que são usadas para corresponder apenas um dos caracteres listados.
- Permite selecionar um dos caracteres dentro dos colchetes, não importando a ordem.

Classes de Caracteres mais usadas

[0-3] Faixa que equivale a [0123]
[a-k] equivale a [abcdefghijk]
[A-C] equivale a [ABC]
[A-Ca-k] equivale a [ABCabcdefghijk]
Expressões POSIX:
[[:alpha:]] equivale a [a-zA-Z]
[^[:alpha:]] Negação de caracteres alfabéticos
[[:upper:]] equivale a [A-Z]
[[:lower:]] equivale a [a-z]
[[:digit:]] equivale a [0-9]
[[:alnum:]] equivale a [0-9a-zA-Z]
[[:space:]] qualquer espaço e branco, incluindo tabulações

Alguns exemplos de Classes de Caracteres

m[ae]u Retorna mau ou meu
[aeiou] Retorna vogais
m[ae][^aeiou] Retorna ma ou me seguido de uma consoante (mas, mal, etc.)

Negação de classes de caracteres: ^
[^aeiou] Nenhum dos caracteres do conjunto.
[^a-zA-Z] Tudo, exceto letras do alfabeto.

Metacaracteres dentro de classes

Muitos metacaracteres dentro de conjuntos de caracteres não precisam de escaping:
[abc.] Qualquer caractere do conjunto, incluindo o ponto .

As exceções são as seguintes (necessitam da barra \):
 ^ \
Ex.: Procurando pelas strings vetor(3) vetor[5]
grep vetor[([][0-9][])]

Metacaracteres - Âncoras

Início de linha
Para representar uma expressão no início de uma linha usamos o caracter ^
grep "^abc" arquivo #Retorna todas as linhas que possuam strings que começam com abc

Fim de linha
Para representar uma expressão no final de uma linha usamos o caracter $
grep "abc$" arquivo #Retorna todas as linhas com strings que terminam com abc

Metacaracteres

Contar linhas vazias (^$)
grep -c "^$" arquivo #Conta as linhas vazias no arquivo.

Repetição de caracteres

Metacaracter	Significado
*	Item precedente zero ou mais vezes
\+	Item precedente uma ou mais vezes
?	Item precedente zero ou uma vez

Repetição de caracteres - *

Zero ou mais ocorrências
Usamos o caracter * para especificar 0 ou mais ocorrências do caracter anterior.

Exemplos:
grep "etc1*" #Retorna as strings que contém etc seguidas ou não do número 1, ou 11, ou 111...

laranjas* #retorna laranja, laranjas ou laranjass

Repetição de caracteres - \+

Uma ou mais ocorrências
Usamos o caracter \+ para especificar 1 ou mais ocorrências do caracter anterior.
grep "etc1\+" #Retorna as strings que contém etc seguidas do número 1, ou 11, ou 111...

colchetes\+ #retorna apenas colchetes ou colchetess; a palavra colchete não seria retornada, pois não tem o "s" final.

"\w\+@" #retorna um ou mais caracteres seguidos de um @; bom para encontrar endereços de email.

Repetição de caracteres - \+

Achar linhas que começam com um ou mais espaços:
"^[ ]\+"

Repetição de caracteres - ?

Zero ou uma ocorrência
Usamos o caracter \? para especificar 0 ou 1 ocorrência do caracter anterior.

Exemplos:
grep "etc1\?" #Retorna as strings que contém etc seguida ou não do número 1.

laranjas\? #retorna laranja ou laranjas, mas não laranjass ou laranjasss, etc.

Repetição Quantificada

{x} Item anterior corresponde exatamente x vezes
{x,} Corresponde ao menos x vezes
{min,max} Corresponde entre min e max vezes

Exemplo:
grep -E r{2} #procura por strings com 'rr' - necessário a opção -E ou usar egrep para repetição quantificada.

Repetição quantificada - exemplos

Achar números de telefone sem código de área:
[0-9]{4}-[0-9]{4}

Achar números de CPF:
\d{3}\.\d{3}\.\d{3}\.-\d{2} #não funcionou c/ grep

Achar sequências de 2 a 5 caracteres seguidas de um espaço:
\w{2,5}\s #funciona com grep -E

{0,} equivale a *
{1,} equivale a \+

Agrupamento de caracteres

Metacaracteres \(e \)
Permitem agrupar partes de uma expressão;
Aplicar operadores de repetição a um grupo
Não podem ser usados dentro de conjuntos de caracteres - tem significado literal lá.

Exemplos (usar egrep ou -E, e sem as aspas):
\(in\)\?dependente # corresponde dependente e independente
\(abc\)\+ #abc ou abcabc ou abcabcabc, etc...

Alternação

Corresponder uma ou outra expressão: |
É como o operador OU
Expressão da esquerda tem preferência.

Exemplo:
egrep "dependente|independente" #acha uma ou outra palavra.
egrep "cas(ado|ada)" #retorna casado ou casada

Caracteres de escape

Para que possamos incluir caracteres especiais nos padrões usamos caracteres de escape.

```
grep "192\.168\.0\.0" /etc/network/interfaces
```

Metacaracteres

\< Início de uma palavra
\> Fim de uma palavra

| Ou (alternação)
- Intervalo
( ) Limitar escopo da alternação
\b Limite de palavras (\bTextoExato\b)

grep

Aplicativo que faz buscas no conteúdo de arquivos por strings especificadas por uma expressão regular (regex).

Global Regular Expression Print.

Sintaxe:

grep [opç] regex [arquivos] #ou

comando | grep [opç] regex #ou

comando | grep regex [opç] | grep regex2

Opções do grep

-c Exibe apenas uma contagem das linha encontradas.

-i Ignora o caso (maiúsculas e minúsculas)

-n Exibe as linhas encontradas com o respectivo número da linha

-v Exibe todas as linhas que NÃO correspondam à regex.

-E Interpreta a regex como uma expressão regular estendida (como egrep)

-r Busca recursiva

--color Mostra a saída colorida.

-A n Mostra a correspondência e as 3 linhas subsequentes

-B n Mostra a correspondência e as 3 linhas anteriores

-C n Mostra a correspondência e as 3 linhas subsequentes e anteriores

Mais exemplos com linhas

grep ^[A-Z] #linhas começando letras maius.
grep ^[^A-Z] #linhas não começando com letra maius.
grep ^[0-9] #linhas começando com núm. 0 a 9
^[[:alpha:]] #linhas que comecem com letras
^[[:upper:]] #linhas que comecem com letras maiúsculas
[[:digit:]]$ #linhas que finalizam com números

Mais exemplos

grep "[fF]ábio" #encontra tanto as strings fábio quanto Fábio.

grep "^[aeiou]" arquivo #Lista apenas as linhas iniciadas com vogais no arquivo.

grep "^[^aeiou]" arquivo #Lista apenas as linhas iniciadas com consoantes ou números no arquivo. (o ^ interno nega)

grep "[^aeiou]$" #lista apenas as linhas terminadas com consoantes ou números.

Mais exemplos

grep -e 'e[a]' #Procura letra 'e' seguida de um 'a'

grep -e 'e[^a]' #Procura letra 'e' não seguida de um 'a'

grep '[1-9]$' #Procura linhas com final entre 1 e 9

grep '[1-9]\$' #procura linhas com um $ final (\$)

egrep '150{3}' #Procura 15 seguido por 3 zeros.

Mais exemplos

'(azul|verde) escuro' #retorna azul escuro ou verde escuro
'azul|verde escuro' #retorna azul ou verde escuro
'(150){3}' #retorna 150150150 (3x)
'[:digit:]' #Qualquer dígito numérico
'[:digit:][:digit:][:digit:]' #Três números em sequência
'[:digit:]{3}' #idem anterior (mais limpo)

Mais exemplos

'Mo{2}ca|Ipiranga' #Mooca ou Ipiranga
'(Mo){2}ca|Ipiranga' #Momoca ou Ipiranga

grep fabio /etc/passwd #procura a string fabio no arquivo /etc/passwd

grep -r 'dhcp' /etc/ #procura a string dhcp em todos os arquivos do diretório /etc

grep -w 'inet' /etc/network/interfaces #procura a palavra inet no arquivo interfaces

egrep -w 'static|dhcp' /etc/networl/interfaces #procura as palavras static e dhcp no arquivo interfaces

Mais exemplos

grep "texto" arq* #procura a string texto dentro de todos os arquivos que começam com arq

grep -rw 'dhcp' /etc/ #procura a palavra dhcp em todos os arquivos do diretório /etc

grep -A 3 -i "exemplo" arquivo #procura a palavra exemplo e retorna também as 3 linhas subsequentes.

grep -B 3 -i "exemplo" arquivo #procura a palavra exemplo e retorna também as 3 linhas anteriores.

grep -C 3 -i "exemplo" arquivo #procura a palavra exemplo e retorna também as 3 linhas anteriores e as 3 linhas subsequentes.

Combinando greps e outros comandos

```
grep "^[^aeiou]" arquivo | grep "^[[:alpha:]]"
# traz apenas linhas iniciadas com consoantes (sem números no início)

ls -l /dev | egrep '(s|h)d[a-z]'    # mostra os nomes dos dispositivos de HDs

find /home/fabio -name "M*" | grep "Mú"
```

Utilitário sed

Stream Editor, ou Editor de Streams
Programa de filtragem usado para automatizar tarefas de edição repetitivas de textos ou processamentos de pipes.

Sintaxe:
sed [opções] 'comando' [arquivo]
sed [opções] -f script [arquivos]

sed - opções

-e comando Essa opção especifica que o argumento na sequência é um comando do sed; opcional ao usar apenas um comando.

-f arquivo Arquivo é um script do sed

-g Trata todas as substituições como globais.

-i Editar arquivos no próprio local

Comandos do sed

d Apaga linhas

s Faz substituições; tem sua sintaxe própria:

s/padrão/substituição/flags

Flags usadas com o comando s:

g Substitui todas as instâncias de padrão

n Substitui a n-ésima instância de padrão

y Traduz caracteres.

Exemplos do sed

Apagar as linhas de 2 a 6 do arquivo:
sed -i '2,6d' arquivo

Apagar as linhas comentadas no arquivo:
sed -i '/^#/d' arquivo #aplicou regex

Traduzir caracteres:
sed -i y/ac/fg/ #a vira f e c vira g

Exemplos do sed

Escrever % em todas as linhas vazias
sed -i 's/^$/%/' arquivo

Apagar linhas em branco:
-i /^$/d

Trocar todas as ocorrências de fabio por ana:
-i s/fabio/ana/g