

Ministério da Educação
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca
UNED Nova Friburgo
Curso Técnico em Informática Integrado ao Ensino Médio

Processamento de texto em terminal e pesquisas

Sistemas Operacionais



Prof. Bruno Policarpo Toledo Freitas
bruno.freitas@cefet-rj.br

Objetivos

- Definir e criar expressões regulares
- Apresentar os principais comandos de processamento de texto em terminal GNU/Linux
 - grep , awk, sed
- Apresentar comando de busca no terminal *find*
- Utilizar formas básicas dos comandos avançados para tarefas elementares
- Aprimorar a utilização do terminal GNU/Linux

Introdução a expressões regulares

- **Expressões regulares são uma “linguagem” para descrever padrões textuais**
 - Não envolve sintaxe
 - Não envolve semântica
- **Ferramenta muito poderosa para resolver problemas de programação**
 - Muitas aplicações de terminal GNU/Linux e (praticamente) todas as linguagens de programação oferecem suporte a expressões regulares

Expressões regulares

(e)grep

- **Busca por textos definidos por expressões regulares em arquivos**
- **Normalmente usado com a extensão *egrep***
- **Utilização padrão:**

(e)grep 'expressao' arquivo

Mostra as linhas onde ocorreram o padrão pesquisado, marcando-o

Expressões regulares

(e)grep - Parâmetros úteis

- **Retornar apenas a parte encontrada:**

(e)grep -o 'expressao' arquivo

Mostra apenas o padrão encontrado

- **Retornar a quantidade de matches:**

(e)grep -c 'expressao' arquivo

- **Retorna apenas se encontrou ou não**

(e)grep -q 'expressao' arquivo

Expressões regulares (e)grep

- **Arquivo de texto**

```
Abracadabra  
Bracadabr  
Racadab  
Acada  
cad  
a  
Hocus pocus  
Avada kedrava
```

Expressões regulares

Padrões simples

- **Padrões são qualquer sequência de caracteres fixa**
- **Exemplos:**
 - 1) Todas as letras “a”
 - ‘a’
 - 2) Todas as ocorrências da palavra “abra”
 - ‘abra’
 - 3) Todas as ocorrências da palavra ‘ocus’
 - ‘ocus’

Expressões regulares

Conjuntos de caracteres

- **Designam qualquer caractere dentro de um conjunto**
- **Denotado pelas chaves []**

Expressões regulares

Conjuntos de caracteres

- **Exemplos:**

- Todas as palavras que contém vogais
 - [aeiou]
- Todas as palavras que contém as letras 'o' e 'u'
 - [ou]
- Todas as palavras que começam com 'a' e são seguidas por 'b' ou 'c'
 - a[bc]
- Todas as palavras que começam com H ou P e terminam com 'ocus'
 - [HP]ocus

Expressões regulares

Conjuntos de caracteres

- **Muitos conjuntos são utilizados com muita frequência e possuem sequencias padrão. Exemplos:**
 - . : qualquer caractere
 - [:digit:] : dígitos
 - [:alpha:]: letras (maiúsculas ou minúsculas)
 - [:alnum:] : alfanuméricos (letras + dígitos)
 - [:space:] : caracteres “invisíveis”
- **Para mais conjuntos: man e grep**

Expressões regulares

Quantificadores

- Designam quantas vezes o padrão se repete
- Os quantificadores são usados após definir um padrão
- Padrões com mais de 1 caractere são agrupados com ()
- Tipos:
 - Opcional, e no máximo 1 vez: ?
 - 0 ou mais vezes: *
 - Pelo menos uma vez: +
 - {mínimo,máximo}

Expressões regulares

Quantificadores

- **Todas as palavras que contém a letra 'a' seguida de 'b' zero ou mais vezes**

`ab*`

- **Todas as palavras que contém 'ab' seguido de 'r' zero ou mais vezes**

`abr*`

- **Todas as palavras que contém a palavra 'abr' pelo menos uma vez**

`abr+`

- **Todas as palavras que contém a palavra 'abra', podendo ser seguidas ou não de 'cada'**

`(abra)(cada)*`

- **Todas as palavras que começam com 'abra' e terminam com 'abra'**

`(abra).*(abra)`

Expressões regulares

Quantificadores

- **Números com 1 ou mais dígitos**

`[[:digit:]]{1,}`

- **Números com 1 até 2 dígitos**

`[[:digit:]]{1,2}`

- **String sem número ou com número de até 2 dígitos**

`[[:digit:]]{,2}`

Expressões regulares

Restrições de início e fim

- Usadas quando queremos pesquisar padrões que se encaixem *exatamente no início ou exatamente no fim*
- Restrição de início:
^
- Restrição de fim:
\$

Expressões regulares

Restrições de início e fim

- **Todas as linhas que começam exatamente com 'abra'**
 - `egrep '^abra'`
- **Todas as linhas que terminam exatamente com 'abra'**
 - `egrep 'abra$'`

Expressões regulares



<https://www.facebook.com/MITCSAIL/posts/10158877210852710>

find

- **Realiza busca por arquivos e pastas**
- **Usos comuns do programa são simples de entender e bastante utilizados**
- **Porém, permite fazer buscas e executar ações bastante complexas**
 - Muito útil em scripts
 - Muito útil com pipes

find

- **Uso básico:**

find diretório -tipo_de_busca parâmetros

- Tipo de busca mais comum é -name

- **Exemplos:**

- Buscar por todos os arquivos nomeados passwd em /etc

- find /etc -name passwd

- Buscar por todos os arquivos .conf em /etc

- find /etc -name *.conf

awk

Exemplos de utilização

- **Printar apenas a coluna 1 do arquivo:**

```
awk '{ print $1 }' arquivo
```

- **Printar apenas a coluna 1 do arquivo, delimitando com ':'**

```
awk -F':' '{ print $1 }' arquivo
```

- **Printar a quantidade de colunas da linha**

```
awk -F':' '{ print NF }' arquivo
```

- **Printar apenas a coluna 1 das linhas 2 a 10:**

```
awk '{ if (NR>=2 && NR <=10 ) print $1 }' arquivo
```

- **Printar a soma das colunas \$2 e \$3**

```
awk '{ print $2+$3 }' arquivo
```

awk



<https://www.facebook.com/nixcraft/posts/4076357845710896>

sed

- ***Stream editor***
- **Linguagem de *scripting* para edição de texto em linha de comando**
- **Muito útil para *shell scripts***
 - Apagar linhas de arquivos
 - Editar texto
 - Substituir texto
 - Isso tudo sem precisar de inputs do usuário

sed

Exemplos

- **Apagar as linhas que contenham ocorrências da palavra *abra*:**

`sed '/abra/d' arquivo`

- **Apagar o conteúdo entre as linhas 2 e 3 de um arquivo:**

`sed '2,3d' arquivo`

- **Substituição de palavras por expressões regulares:**

`sed 's/abra/cada/g' arquivo`

Referências

- **FILHO, J. E. M. *Descobrendo o Linux: Entenda o sistema operacional GNU/Linux*. São Paulo, Novatec Editora, 3a. Edição**
 - Cap. 17, Seção 17.1.8 (find)
 - Cap. 25, Seções 25.3 (expressões regulares), 25.10.8 (sed)

Referências

- **Introduction to the command line**
 - awk
 - sed
 - Expressões regulares

Referências

- **man pages**

- grep
 - tutorial completo de expressões regulares
- find
- awk
- sed

Referências

- **ARS TECHNICA. Linux/BSD command line wizardry: Learn to think in sed, awk, and grep**