Introdução à Ciência da Computação

Gerenciamento de Texto em Arquivos

Prof. Iago Augusto de Carvalho

iago.carvalho@unifal-mg.edu.br

1

Buscar em Arquivos de Texto Usando Expressão Regular

• Expressão Regular – Exemplos:

• [A-Za-z] Qualquer letra;

• [^0-9A-Za-z] Qualquer símbolo (exceto números e letras);

• [A-Z][a-z]\* Uma letra maiúscula, seguida de zero ou mais letras minúsculas; • [A-Z][a-z]+ Uma letra maiúscula, seguida de uma ou mais letras minúsculas; • [A-Z][a-z]? Uma letra maiúscula, seguida de zero ou uma letra minúscula;

2

Buscar em Arquivos de Texto Usando Expressão Regular

• Comando grep

• Usado para buscar em arquivos ou na entrada padrão por linhas que contenham uma string que corresponda à expressão regular montada;

• Sintaxe:

• grep [opções] expressão\_regular [arquivos]

• Algumas opções

• -v Exibe todas as linhas que não batam com a regex;

• -i Ignora a distinção entre maiúsculas e minúsculas;

• -c exibe a contagem das linhas encontradas, mas não as linhas

3

Buscar em Arquivos de Texto Usando Expressão Regular

• Exemplos:

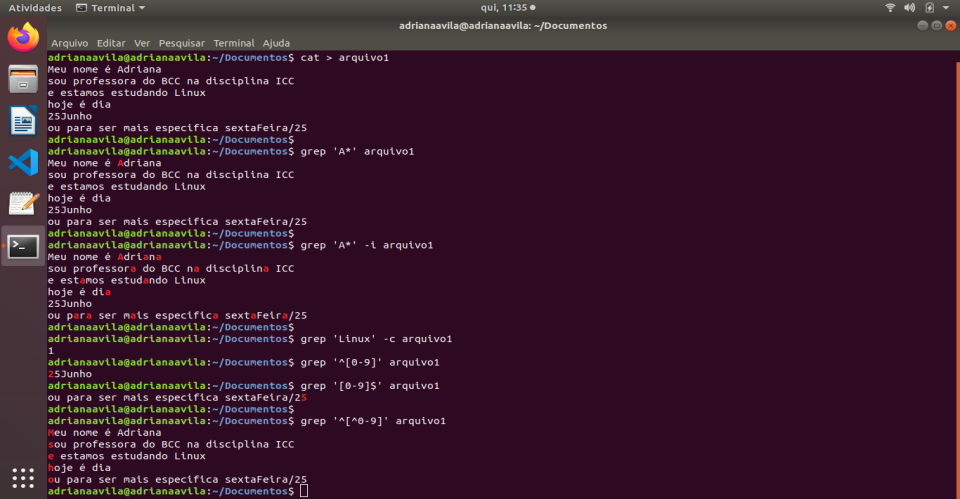
• grep ‘Linux’ arquivo1

•grep ‘^[0-9]‘ (começa com número)

• grep ‘[0-9]$‘ (termina com número)

• grep ‘^[^0-9]‘ (começa com não número)

4

5

Processar streams de texto usando filtros

O Linux dispõe de vários comandos cujo objetivo é servirem como filtros, modificando a saída de textos de forma mais útil;

A aula de hoje mostra com exemplos vários comandos dessa natureza; 6

Processar streams de texto usando filtros

• Comando cut

• Corta (exibindo colunas) de um ou mais arquivos;

• O arquivo original não é modificado;

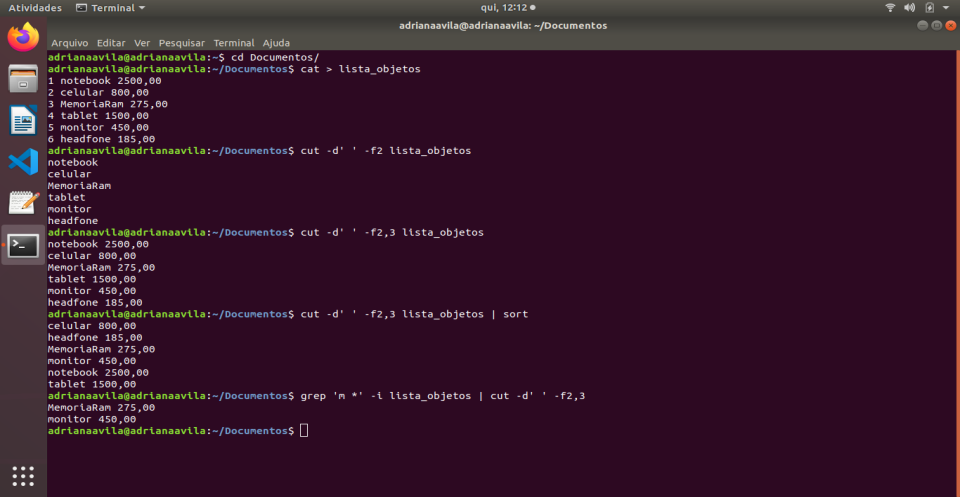
• Exemplo:

• cut –d’ ‘ –f2,3 lista\_objetos

• Onde:

• -d delimitador: configura o delimitador que separa uma coluna da outra. O padrão é o Tab;

• -f número: imprime a coluna que corresponde ao número passado. 7

8

Processar streams de texto usando filtros

• Comando expand

• Converte tabs em espaços;

• Por padrão, assume-se que um tab equivale a 8 espaços;

• Sintaxe:

• expand [opções] [arquivos] > [arquivo\_saida]

• Algumas opções:

• -t *número* especifica o número de espaços de tab, ao invés de 8; • -i converte apenas no início das linhas;

9

Processar streams de texto usando filtros

• Comando unexpand

• Converte espaços em tabs;

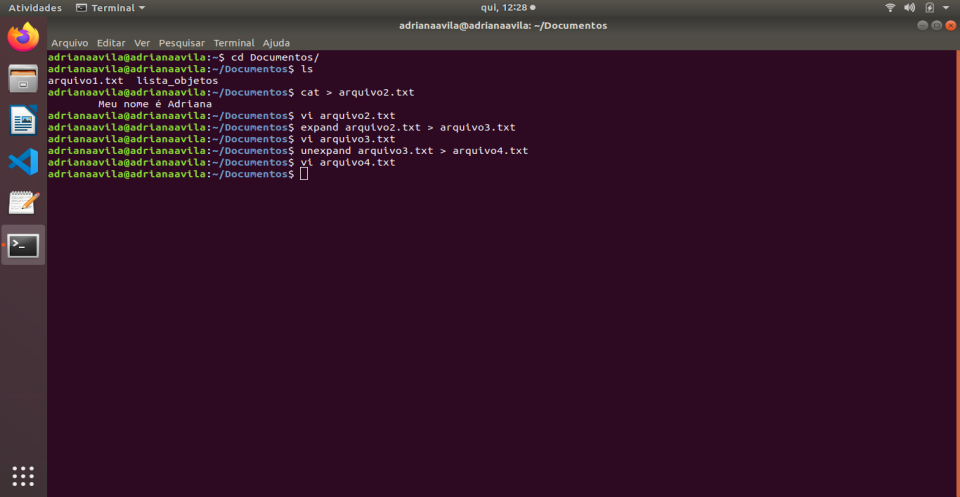
• Por padrão, o unexpand só funciona em espaço no início das linhas; • Sintaxe:

• unexpand [opções] [arquivos] > [arquivo\_saida]

• Algumas opções:

• -a Converte todos os espaços, não apenas os iniciais;

10

11

Processar streams de texto usando filtros

• Comando head

• Exibe as primeiras linhas de um ou mais arquivos (com padrão igual a 10 linhas); • Sintaxe

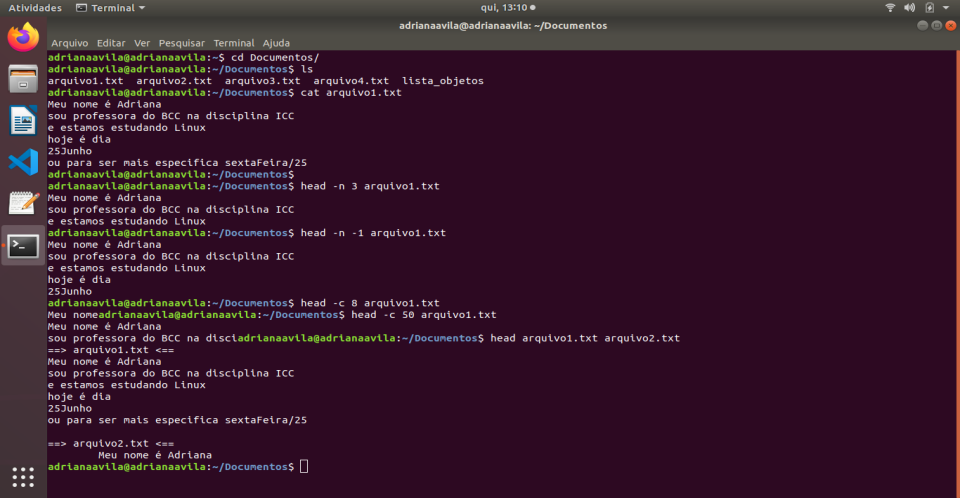
• head [opções] [arquivos]

• Algumas opções:

• -c número: exibe na tela, o texto, conforme o número de bytes passado;

• -n número: exibe na tela, a quantidade de linhas do texto, conforme o número passado.

12

13

Processar streams de texto usando filtros

• Comando tail

• Exibe as últimas linhas de um ou mais arquivos (com padrão igual a 10 linhas); • Sintaxe

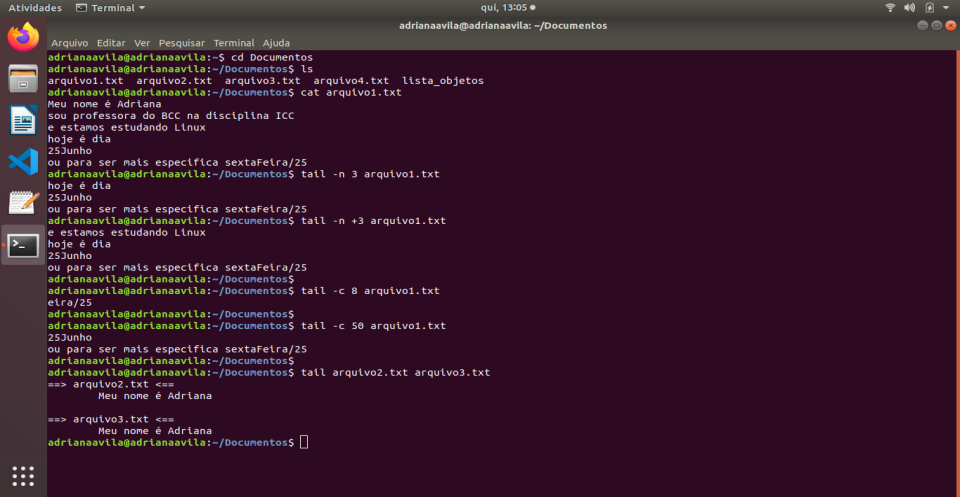
• tail [opções] [arquivos]

• Algumas opções:

• -c número: exibe na tela, o texto, conforme o número de bytes passado;

• -n número: exibe na tela, a quantidade de linhas do texto, conforme o número passado.

14

15

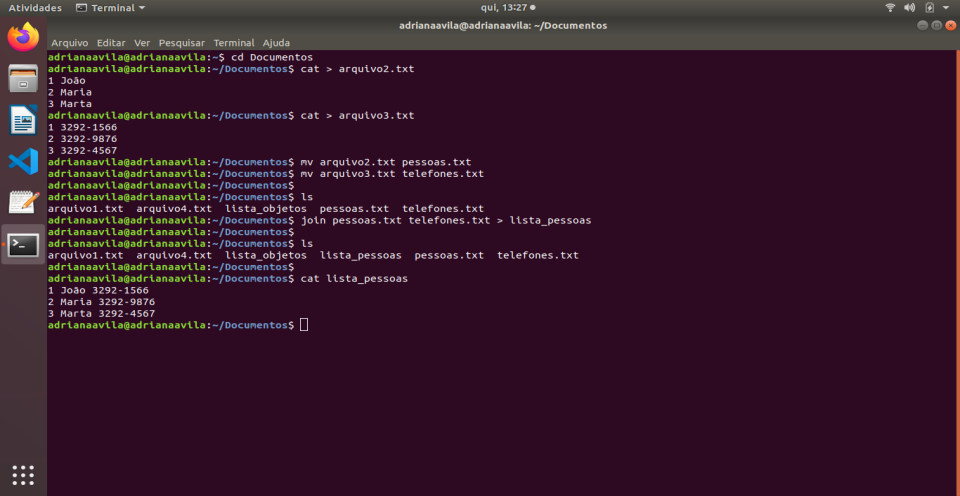
Processar streams de texto usando filtros

• Comando join

• Executa um tipo de join semelhante à banco de dados, mas em arquivos; • Esse comando une as linhas de ambos os arquivos que tenham um índice em comum. • Sintaxe

• join [arquivo1][arquivo2]

16

17

Processar streams de texto usando filtros

• Comando paste

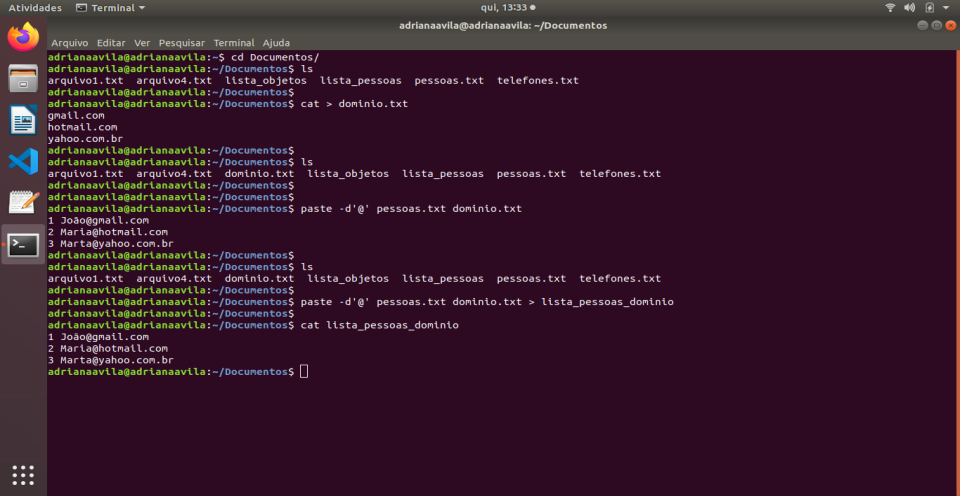
• Cola as linhas correspondentes de um ou mais arquivos em colunas verticais • Sintaxe

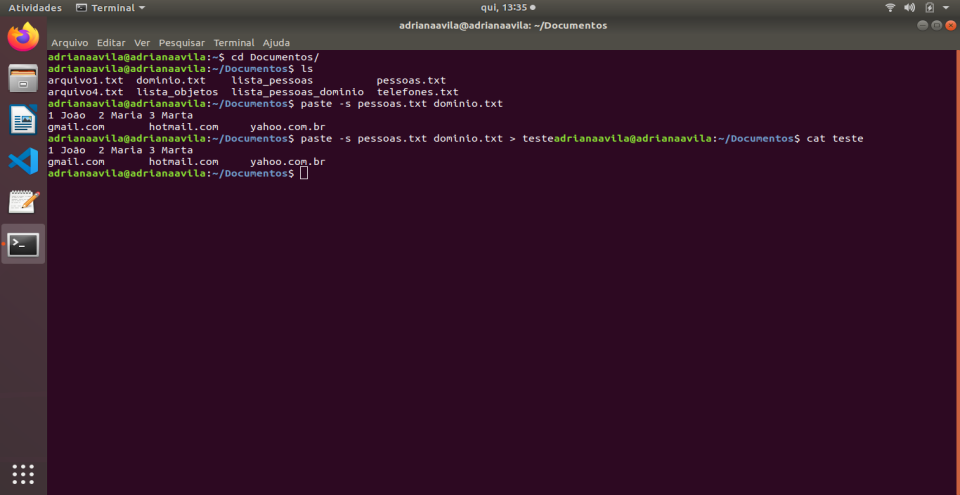
• paste [opções] [arquivos]

• Algumas opções:

• -d’caractere’: separa as colunas com o caractere dentro das aspas simples; • -s: concatena todo o conteúdo de um arquivo com uma linha para cada arquivo.

18

19

20

Processar streams de texto usando filtros

• Comando sort

• Retorna para stdout a entrada classificada alfabeticamente; • Sintaxe

• sort [opções] [arquivos]

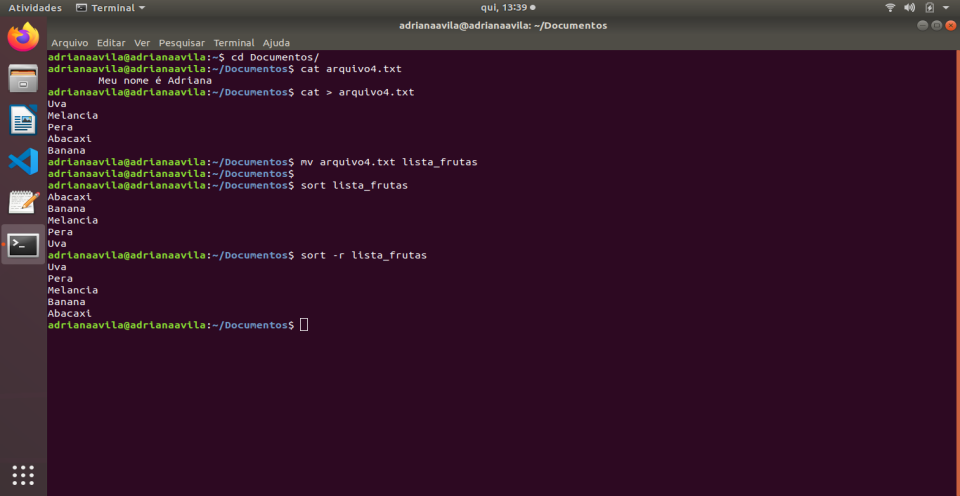
• Exemplo:

• sort lista\_frutas

• sort -r lista\_frutas

Onde -r retorna em ordem inversa;

21

22

Processar streams de texto usando filtros

• Comando split

• O comando split (dividir) é usado para dividir grandes arquivos em n-arquivos menores. • Os arquivos de saída são gerados de acordo com o número de linhas do arquivo de entrada. • O padrão é dividir o arquivo a cada 1000 linhas.

• Sintaxe

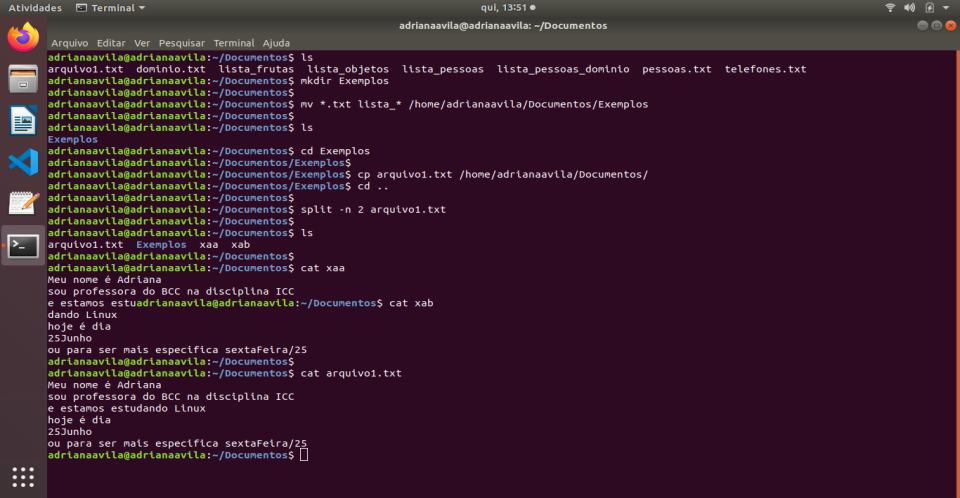
• split [opção] [arquivo de entrada] [arquivo de saída]

• Exemplo:

• split -n 2 arquivo1.txt

• Algumas opções:

• -n: onde n é o número de arquivos que irão dividir o arquivo de entrada 23

24

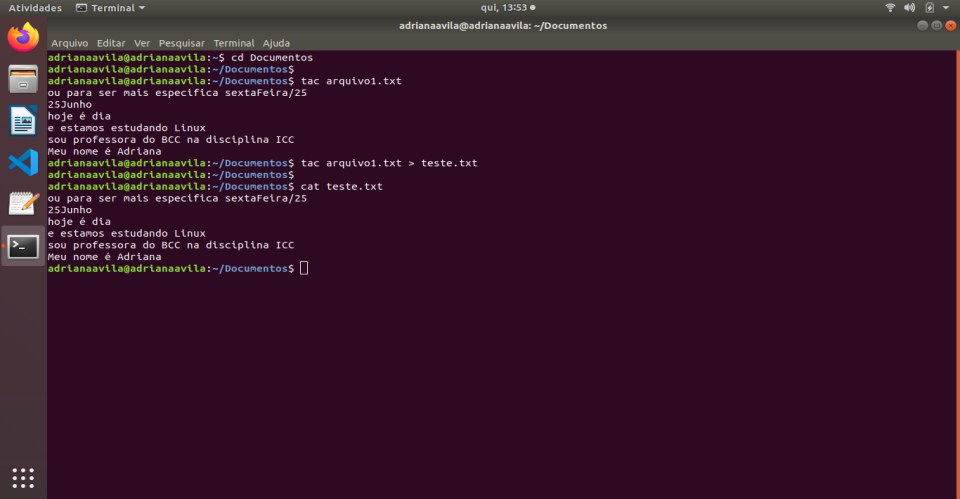
Processar streams de texto usando filtros

• Comando tac

• Semelhante ao cat, mas envia os arquivos para stdout com as linhas em ordem inversa; • Sintaxe

• tac [arquivo]

25

26

Processar streams de texto usando filtros

• Comando wc

• Exibe a contagem de linhas, palavras e caracteres para arquivos;

• Sintaxe

• wc [opções] [arquivos]

• Algumas opções:

• -c exibe apenas a contagem de caracteres;

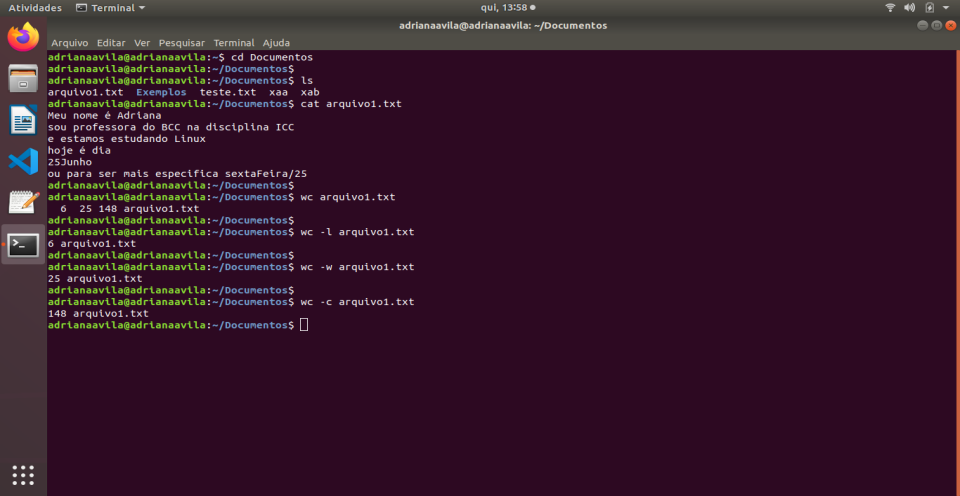
• -l exibe apenas a contagem de linhas;

• -w exibe apenas a contagem de palavras;

• Exemplo:

• wc arquivo1.txt

27

28

Referências

Slides adaptados do Prof. Flavio Barbieri Gonzaga e do curso Introdução ao SO Linux – Renata Spolon e Gabriel Martinez

PRITCHARD, S.; PESSANHA, B. G.; LANGFELDT, N.; STANGER, J.; DEAN, J. 2007. **Certificação Linux LPI Rápido e Prático. Guia de Referência nível 1: Exames 101 e 102**. 2ª Ed. Rio de Janeiro: Editora Alta Books.

29