Máster de Bioinformática y Biología Computacional Universidad Autónoma de Madrid
Procesado y Manejo Masivo de Datos

Elena Aguayo Jara
30/10/2022

Exercises

PART 2

MORE ON FILTERS

Ex 1.

- List all the processes of the system: ps -ef
- List all the processes for the current user: ps-ef | grep ' elena_aguayo'
- List the 10 newest processes for the current user: ps -ef | grep elena | sort -k5,5 | tail -10
- Display the manual pages por the grep command using at most 100 chars per line: man grep | fmt -100
- Same as before but substituting any digit by an asterisk(*): man grep | fmt -100 -s | tr '[0-9]' | '*'
- Same as before but sending the result to a file: man grep | fmt -100 -s | tr ' [0-9]' | '*' > file.txt

Ex 2. Extract the columns 6 and 4 (in this order) of file *adult.data*, not necessarily with a single line command

cut adult.data -d "," -f 6,4

Ex 5. Repeat the first example of previous slide, but display only the key and the field 2 of the second file (*f17.txt*)

join -t "," -1 2 -2 2 -a 2 f13.txt f17.txt

Ex 6. Write shell commands to perform the following actions:

- Create a file *a.txt* with 20 random numbers between 0 and 100, without repetition; each number must be in a different line

  echo {0..100} | tr ' ' '\n' | shuf -i 0-100 | head -20 > a.txt)

- Create a second file *b.txt* also with 20 random numbers between 0 and 100, without repetition; each number must be in a different line

  echo {0..100} | tr ' ' '\n' | shuf -i 0-100 | head -20 > b.txt

- Display the numbers that appear in both files

  comm -3 a.txt b.txt

Ex 7. Write a shell script that accepts 3 arguments: *f*, *a*, *b*. The argument *f* is the name of a *csv* file; the arguments *a* and *b* are integer numbers. The script must display the list of values that appear in (both) columns *a* and *b* of file *f*.

Escribo en el editor gedit lo siguiente:

```
#!/bin/bash
f=$1
a=$2
b=$3
cut -d ',' -f $a,$b $f
```

TEXT EDITORS

Ex 8. How can you open the vi editor placing the cursor at the end of the file?
vi + file.txt

Ex. 9. How can you undo the last change in the vi editor?
u

Ex. 10. What is the meaning of the command: %s/hola/adios/gc in the vi editor?
Sustituye 'holas' cada vez que aparezca por 'adios' preguntándonos si estamos seguros.

Ex. 11. In vi, which commands do you use to copy the current line just below?
Y o yy (para copiar)

p (para pegar).

PART 3

REGULAR EXPRESSIONS AND GREP

Ex 1. Write a grep regular expression to extract the lines that contain only numbers

```
grep '^[0-9][0-9]*[0-9]$ ' adult.data
```

Ex 2. Write a grep regular expression that matches strings in the file *splice.data* with no T and an odd number of Gs

```
cut splice.data -d ',' -f3| grep -E '\b[AC]G([AC]*G[AC]*G)[AC]*$'
```

Ex 3. Write a grep regular expression that matches e-mail addresses of the form *x.y@t.z*, where:

- *x* and *y* are non-empty strings that may contain lowercase letters, digits or the underscore symbol (_), but must start with a letter
- *t* is a non-empty string that contains only lowercase letters
- *z* is either *es* or *com*

```
grep -E "^[a-zA-Z]([a-zA-Z]*[0-9]*(_)*)+(\.)[a-zA-Z]([a-zA-Z]*[0-9]*(_)*)*@[a-z]+(\.com|\.es)$"
```

Ex. 4. Write a single line command that, using brace expansion, creates the list of directories *mmm-yy*, where *mmm* is a month (jan, feb,...) and *yy* is a year (16, 17, 18)

echo {jan,feb,march,april,june,july,aug,sept,oct,nov,dec}-{16,17,18}

Ex. 5. Extract the columns 6 and 4 (in this order) of file *adult.data* using a single line command

cut adult.data -d ',' -f 6,4

PART 4: THE SED COMMAND

Which is the outpuf of the following commands?

- seq 1 20 | sed -n 'n;n;p'

  3, 6, 9, 12, 15, 18

Secuencia de números del 1 al 20 separados por saltos de línea, donde -n no imprime el resultado, se imprime cada 3 saltos de línea.

- seq 1 20 | sed -n '/[26]/{n;n;p}'

  4, 8, 14, 18

Hace lo mismo pero cada vez que encuentra 2 ó 6, por ejemplo, al encontrar 2, hace 2 ,3 4 e imprime el 4. Solo ejecuta en esas líneas por poner llaves cuando coinciden con el patrón.

Ex 1. What is the output of the last command if we remove the braces?

Output: 2, 4, 6, 8, 10, 12, 14, 16, 18, 20.

Al quitar las llaves, ejecuta el comando en todas las líneas, no solo cuando encuentra el patrón.

Ex 2. Write a sed command to display lines from 100 to 200 (inclusive) of file *adult.data*

sed -n '100,200p' adult.data

Ex 3. Modify the previous command so that it stops processing the file when the last line is printed

sed -n '100, $p' adult.data

PART 5: THE AWK COMMAND

Ex 1. Write an awk program that prints the age (field 1), education (field 4), gender (field 10), marital status (field 6) and working hours per week (field 13) for all records in the file *adult.data* where the country (field 14) is *United-States*

- The fields must be displayed in the previous order
- They must be separated by a semicolon (;)

```
awk  'BEGIN{FS=", "; OFS = ";"} $14~/United-States/{print $1,$4,$10,$6,$13}' adult.data
```

Ex 2. Modify the previous program so that it prints the following header before any other output

```
awk 'BEGIN {print "Age;Education;Gender;Marital-status;Hours-per-week"; FS=", "; OFS = ";"}{print age,$1,$4,$10,$6,$13}' adult.data
```

Ex 3. Write an awk program that processes the file *adult.data* and prints the mean number of hours worked by men and women

```
awk '{sum+=$13; n++} END {if (n > 0) print sum/n;}' adult.data
```

Ex 4. Write an awk program that processes the file *adult.data* and prints a list with the proportion of people that works less than 10 hours per week for each educational level.

Primero guardamos la lista en una variable, en esta caso llamada *var.*

```
awk  'BEGIN{FS=", "; OFS = ";" var=$4} {print $4}' adult.data  |sort | uniq > work.txt
```

```
awk 'BEGIN{max =10 ;Hours = 0;FS=", "}$4=="Hours" {if ($13<max) Hours++} END {print(Hours);uniq $4}' adult.data
```

Ex 5. Use awk or sed to process the file *splice.data* and print all the records where the sequence contains a string that starts with GAA, ends with CTA and is longer than 10 characters

```
awk ' /(GAA)+(A|C|G|T){5, }+((CTA)$)/ {print $3}'   splice.data
```