Tarea: TC3_Forloop

1.10 Exercises

1.10.1 Next Generation Sequencing Data

In this exercise we work with next generation sequencing (NGS) data. Unix is excellent at manipulating the huge FASTA files that are generated in NGS experiments.

FASTA files contain sequence data in text format. Each sequence segment is preceded by a single-line description. The first character of the description line is a "greater than" sign (>).[15]

The NGS data set we will be working with was published by Marra and DeWoody (2014), who investigated the immunogenetic repertoire of rodents. You will find the sequence file `Marra2014_data.fasta` in the directory `CSB/unix/data`. The file contains sequence segments (contigs) of variable size. The description of each contig provides its length, the number of reads that contributed to the contig, its isogroup (representing the collection of alternative splice products of a possible gene), and the isotig status.

1. Change directory to `CSB/unix/sandbox`.
2. What is the size of the file `Marra2014_data.fasta`?[16]
3. Create a copy of `Marra2014_data.fasta` in the `sandbox` and name it `my_file.fasta`.
4. How many contigs are classified as `isogroup00036`?
5. Replace the original "two-spaces" delimiter with a comma.
6. How many unique isogroups are in the file?
7. Which contig has the highest number of reads (`numreads`)? How many reads does it have?

1.10.1 Capturas de la Resolución del ejercicio.

1. Cambio de Directorio:

```
@Karlitoz2001 →/workspaces/CSBCarlitos (master) $ cd ..
@Karlitoz2001 →/workspaces $ cd CSBCarlitos/unix/sandbox/
@Karlitoz2001 →/workspaces/CSBCarlitos/unix/sandbox (master) $ ls
 Carlitos.fasta  'Papers and reviews'   TC3.sh
@Karlitoz2001 →/workspaces/CSBCarlitos/unix/sandbox (master) $
```

2. ¿Cuál es el tamaño del archivo Marra2014_data.fasta?

```
@Karlitoz2001 →/workspaces/CSBCarlitos/unix/sandbox (master) $ ls -lh ../data/Marra2014_data.fasta
-rw-rw-rw- 1 codespace root 553K May  9 05:57 ../data/Marra2014_data.fasta
@Karlitoz2001 →/workspaces/CSBCarlitos/unix/sandbox (master) $
```

3. Copiar el archivo de Marra2014 a sandbox con el nombre my_file.fasta

```
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ cp ../data/Marra2014_data.fasta Carlitos.fasta
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ ls
 Carlitos.fasta  'Papers and reviews'   TC3.sh
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $
```

4.- ¿Cuántos isogrupos hay ?

```
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ grep isogroup00036 Carlitos.fasta | wc -l
16
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $
```

5.- Reemplazar el delimitador con dos espacios y con una coma.

```
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ cat Carlitos.fasta | tr -s "  " "," > Carlitos.tmp
cat: Carlitos.fasta: No such file or directory
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ ls
 Carlitos.tmp  'Papers and reviews'   TC3.sh
```

```
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ mv Carlitos.tmp Carlitos.fasta
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ ls
 Carlitos.fasta  'Papers and reviews'   TC3.sh
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $
```

6.- ¿Cuántos isogrupos hay en el archivo?

```
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ grep isogroup Carlitos.fasta | cut -d ',' -f 4 | sort | uniq | wc -l
955
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ grep isogroup00036 Carlitos.fasta | cut -d ',' -f 4 | sort | uniq | wc -l
16
```

7.- Cual es el mayor número de lecturas Contig.

```
@Karlitoz2001 ➜/workspaces/CSBCarlitos/unix/sandbox (master) $ grep '>' Carlitos.fasta | cut -d "," -f 1,3 | sort -t '=' -k 2 -n -r | head -n 1
>contig01115  length=6087  numreads=185  gene=isogroup00030  status=isotig
```

1.10.2 Hormone Levels in Baboons

Gesquiere et al. (2011) studied hormone levels in the blood of baboons. Every individual was sampled several times.

15. See computingskillsforbiologists.com/fasta for more details on the FASTA file format.
16. Note that the original sequence file is much larger! We truncated the file to 1% of its original size to facilitate the download.

1. How many times were the levels of individuals 3 and 27 recorded?
2. Write a script taking as input the file name and the ID of the individual, and returning the number of records for that ID.
3. [**Advanced**][17] Write a script that returns the number of times each individual was sampled.

1.10.2 Capturas de la Resolución del ejercicio.

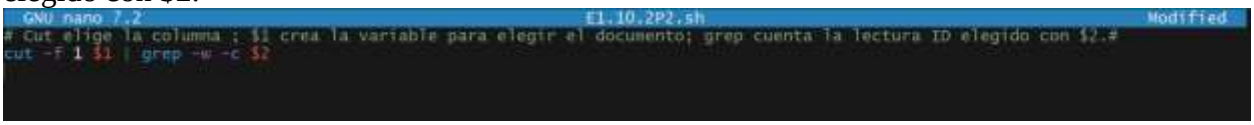
1. ¿Cuántas veces se registraron los niveles de los individuos 3 y 27

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ head -n 5 Gesquiere2011_data.csv
maleID  GC      T
1       66.9    64.57
1       51.09   35.57
1       65.89   114.28
1       80.88   137.81

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ cut -f 1 Gesquiere2011_data.csv | grep -w -c 3
61

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ cut -f 1 Gesquiere2011_data.csv | grep -w -c 27
5
```

2. Escriba un script tomando como entrada el nombre del fichero y el ID del individuo y devolviendo el número de registros para ese ID.

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ echo "Pregunta 2 del Ejercicio 1.10.2"
Pregunta 2 del Ejercicio 1.10.2

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (mast
er)
$ nano E1.10.2P2.sh
```

Cut elige la columna ; $1 crea la variable para elegir el documento; grep cuenta la lectura ID elegido con $2.




```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ bash E1.10.2P2.sh Gesquiere2011_data.csv 4
46

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (mast
er)
$ bash E1.10.2P2.sh Gesquiere2011_data.csv 30
63

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (mast
er)
$ bash E1.10.2P2.sh Gesquiere2011_data.csv 19
3

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (mast
er)
$ bash E1.10.2P2.sh Gesquiere2011_data.csv 1
10
```

3. Escriba un script que devuelva el número de veces que cada individuo fue muestreado.

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ echo "Pregunta 3 del Ejercicio 1.10.2"
Pregunta 3 del Ejercicio 1.10.2

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (mast
er)
$ nano E1.10.2P3.sh
```

```
  GNU nano 7.2                    E1.10.2P3.sh                          Modified
# Se toma la lista de los IDs y se elimina la fila del encabezado.#

ids=`cut -f 1 Gesquiere2011_data.csv | tail -n +2 | sort -n |uniq`

#Para los IDs se genera un bucle.#

for tigre in $ids
do
        numero=`bash E1.10.2P2.sh Gesquiere2011_data.csv $tigre`
        echo "ID:" $tigre "Cantidad:" $numero
done
```

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ bash E1.10.2P3.sh
ID: 1 Cantidad: 10
ID: 2 Cantidad: 2
ID: 3 Cantidad: 61
ID: 4 Cantidad: 46
ID: 5 Cantidad: 28
ID: 6 Cantidad: 7
ID: 7 Cantidad: 5
ID: 8 Cantidad: 17
ID: 9 Cantidad: 4
ID: 10 Cantidad: 21
ID: 11 Cantidad: 26
ID: 12 Cantidad: 23
ID: 13 Cantidad: 16
ID: 14 Cantidad: 1
ID: 15 Cantidad: 40
ID: 16 Cantidad: 31
ID: 17 Cantidad: 3
ID: 18 Cantidad: 4
ID: 19 Cantidad: 3
ID: 20 Cantidad: 4
ID: 21 Cantidad: 12
ID: 22 Cantidad: 5
ID: 23 Cantidad: 36
ID: 24 Cantidad: 35
ID: 25 Cantidad: 35
ID: 26 Cantidad: 22
ID: 27 Cantidad: 5
ID: 29 Cantidad: 33
ID: 30 Cantidad: 63
ID: 31 Cantidad: 1
ID: 32 Cantidad: 3
ID: 33 Cantidad: 1
ID: 34 Cantidad: 16
ID: 35 Cantidad: 5
ID: 36 Cantidad: 39
ID: 37 Cantidad: 38
ID: 38 Cantidad: 1
ID: 39 Cantidad: 3
```

1.10.3 Plant–Pollinator Networks

Saavedra and Stouffer (2013) studied several plant–pollinator networks. These can be represented as rectangular matrices where the rows are pollinators, the columns plants, a 0 indicates the absence and 1 the presence of an interaction between the plant and the pollinator.

The data of Saavedra and Stouffer (2013) can be found in the directory `CSB/unix/data/Saavedra2013`.

1. Write a script that takes one of these files and determines the number of rows (pollinators) and columns (plants). Note that columns are separated by spaces and that there is a space at the end of each line. Your script should return

```
$ bash netsize.sh ../data/Saavedra2013/n1.txt
Filename: ../data/Saavedra2013/n1.txt
Number of rows: 97
Number of columns: 80
```

2. [**Advanced**][18] Write a script that prints the numbers of rows and columns for each network:

```
$ bash netsize_all.sh
../data/Saavedra2013/n10.txt 14 20
../data/Saavedra2013/n11.txt 270 91
../data/Saavedra2013/n12.txt 7 72
../data/Saavedra2013/n13.txt 61 17
...
```

1.10.3 Capturas de la Resolución del ejercicio.

1. Escribe un script que tome uno de estos archivos y determine el número de filas (polinizadores) y columnas (plantas). Tenga en cuenta que las columnas están separadas por espacios y que hay un espacio al final de cada línea. y que hay un espacio al final de cada línea. Su script debe devolver

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix (master)
$ echo "Ejercicio 1.10.3 Pregunta 1."
Ejercicio 1.10.3 Pregunta 1.

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix (master)
$ cd data/Saavedra2013/

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ nano n1.txt

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ nano E1.10.3P1.sh

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ wc -l n1.txt
98 n1.txt

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ head -n 1 n1.txt
1 1 1 0 1 1 1 0 1 0 1 1 0 0 0 1 1 1 1 1 1 1 0 0 0 0 1 0 1 0 1 0 0 1 1 1 1 1 1 0 0 1 0 1 0 0 0 0
 1 1 1 0 1 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ head -n 1 n1.txt
1 1 1 0 1 1 1 0 1 0 1 1 0 0 0 1 1 1 1 1 1 1 0 0 0 0 1 0 1 0 1 0 0 1 1 1 1 1 1 0 0 1 0 1 0 0 0 0
 1 1 1 0 1 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ head -n 1 n1.txt | tr -d " " | wc -c
82

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ head -n 1 n1.txt | tr -d " " | tr -d "\n" | wc -c
81

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ bash E1.10.3P1.sh
Numero de Filas:
98 n1.txt
Numero de Columnas:
81
```

```
  GNU nano 7.2                                                E1.10.3P1.sh
#Para generar un numero de filas#

echo "Numero de Filas:"
wc -l n1.txt

#Generar numero de columnas#

echo "Numero de Columnas:"
head -n 1 n1.txt | tr -d " " | tr -d "\n" |wc -c
```

2. Escribe un script que imprima los números de filas y columnas de cada red:

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ echo "Ejercicio 1.10.3 Pregunta 2"
\
Ejercicio 1.10.3 Pregunta 2
>

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ ls
Buzzard2015_about.txt  E1.10.2P3.sh             Marra2014_about.txt     Pacifici2013_data.csv   miRNA/
Buzzard2015_data.csv   Gesquiere2011_about.txt  Marra2014_data.fasta    Saavedra2013/
E1.10.2P2.sh           Gesquiere2011_data.csv   Pacifici2013_about.txt  Saavedra2013_about.txt

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ cd Saavedra2013/

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ nano E1.10.3P2
```

```
  GNU nano 7.2                                          E1.10.3P2.sh
#Se genera filas y columnas anidado con un for#

for ar in *.txt
do
        filas=`cat $ar | wc -l`
        columnas=`head -n 1 $ar | tr -d " " | tr -d "\n" | wc -c`
        echo "Archivo:" $ar "Filas:" $filas "Columnas:" $columnas
done
```

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ bash E1.10.3P2.sh
Archivo: n1.txt Filas: 98 Columnas: 81
Archivo: n10.txt Filas: 14 Columnas: 21
Archivo: n11.txt Filas: 270 Columnas: 92
Archivo: n12.txt Filas: 7 Columnas: 73
Archivo: n13.txt Filas: 61 Columnas: 18
Archivo: n14.txt Filas: 35 Columnas: 16
Archivo: n15.txt Filas: 38 Columnas: 12
Archivo: n16.txt Filas: 118 Columnas: 25
Archivo: n17.txt Filas: 76 Columnas: 32
Archivo: n18.txt Filas: 13 Columnas: 15
Archivo: n19.txt Filas: 10 Columnas: 17
Archivo: n2.txt Filas: 62 Columnas: 42
Archivo: n20.txt Filas: 18 Columnas: 8
Archivo: n21.txt Filas: 19 Columnas: 46
Archivo: n22.txt Filas: 19 Columnas: 37
Archivo: n23.txt Filas: 179 Columnas: 27
Archivo: n24.txt Filas: 80 Columnas: 29
Archivo: n25.txt Filas: 17 Columnas: 17
Archivo: n26.txt Filas: 82 Columnas: 41
Archivo: n27.txt Filas: 27 Columnas: 6
Archivo: n28.txt Filas: 90 Columnas: 20
Archivo: n29.txt Filas: 61 Columnas: 26
Archivo: n3.txt Filas: 25 Columnas: 37
Archivo: n30.txt Filas: 8 Columnas: 20
Archivo: n31.txt Filas: 28 Columnas: 26
Archivo: n32.txt Filas: 45 Columnas: 22
Archivo: n33.txt Filas: 70 Columnas: 21
Archivo: n34.txt Filas: 79 Columnas: 26
Archivo: n35.txt Filas: 14 Columnas: 9
Archivo: n36.txt Filas: 40 Columnas: 170
Archivo: n37.txt Filas: 44 Columnas: 14
Archivo: n38.txt Filas: 51 Columnas: 100
Archivo: n39.txt Filas: 33 Columnas: 26
Archivo: n4.txt Filas: 101 Columnas: 12
Archivo: n40.txt Filas: 28 Columnas: 19
Archivo: n41.txt Filas: 12 Columnas: 11
Archivo: n42.txt Filas: 42 Columnas: 9
Archivo: n43.txt Filas: 55 Columnas: 30
Archivo: n44.txt Filas: 56 Columnas: 10
Archivo: n45.txt Filas: 36 Columnas: 62
```

3.¿Qué fichero tiene el mayor número de filas? ¿Cuál tiene el mayor número de columnas?

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/S
aavedra2013 (master)
$ echo "Ejercicio 1.10.3 Pregunta 3"
Ejercicio 1.10.3 Pregunta 3

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/S
aavedra2013 (master)
$ nano E1.10.3P3.sh
```

```
  GNU nano 7.2                                  E1.10.3P3.sh
#Se utiliza el anterior codigo, sin embargo se elimina la parte de filas y columnas.#

for ar in *.txt
do
        filas=`cat $ar | wc -l`
        columnas=`head -n 1 $ar | tr -d " " | tr -d "\n" | wc -c`
        echo $ar $filas $columnas
done
```

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ bash E1.10.3P3.sh | head -n 5
n1.txt 98 81
n10.txt 14 21
n11.txt 270 92
n12.txt 7 73
n13.txt 61 18
```

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ echo "Para el mayor numero de filas esta en el .txt"
Para el mayor numero de filas esta en el .txt

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data/Saavedra2013 (master)
$ bash E1.10.3P3.sh | sort -n -r -k 2 | head -n 1
n58.txt 678 91
```

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix
/data/Saavedra2013 (master)
$ echo "el mayor numero de columnas esta en el .txt
"
el mayor numero de columnas esta en el .txt

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix
/data/Saavedra2013 (master)
$ bash E1.10.3P3.sh | sort -n -r -k 3 | head -n 1
n56.txt 110 208
```

1.10.4 Capturas de la Resolución del ejercicio.

1.10.4 Data Explorer

Buzzard et al. (2016) collected data on the growth of a forest in Costa Rica. In the file `Buzzard2015_data.csv` you will find a subset of their data, including taxonomic information, abundance, and biomass of trees.

1. Write a script that, for a given CSV file and column number, prints
   - the corresponding column name;
   - the number of distinct values in the column;
   - the minimum value;
   - the maximum value.

UNIVERSIDAD REGIONAL AMAZÓNICA IKIAM
Carlos Francisco Palacios Almeida
Bioinformática

1. Escriba una secuencia de comandos que, para un archivo CSV y un número de columna dados, imprima

- el nombre de la columna correspondiente;
- el número de valores distintos en la columna;
- el valor mínimo;
- el valor máximo.

```
carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix (master)
$ cd data

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ ls
Buzzard2015_about.txt  E1.10.2P2.sh  Gesquiere2011_about.txt  Marra2014_about.txt   Pacifici2013_about.txt  Saavedra2013/          miRNA/
Buzzard2015_data.csv   E1.10.2P3.sh  Gesquiere2011_data.csv   Marra2014_data.fasta  Pacifici2013_data.csv   Saavedra2013_about.txt

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ echo "EJercicio 1.10.4 Pregunta 1."
EJercicio 1.10.4 Pregunta 1.

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ nano E1.10.4P1.sh

carlo@NSX MINGW64 ~/Desktop/Bioinformática/CSB/unix/data (master)
$ bash E1.10.4P1.sh
Extracción del nombre de la columna
biomass
Número de valores distintos
285
Valor máximo
14897.29471
Valor mínimo
1.048466198
```

MINGW64:/c/Users/carlo/Desktop/Bioinformática/CSB/unix/data

```
 GNU nano 7.2                    E1.10.4P1.sh                          Modified
echo "Nombre de la columna"

cut -d ',' -f 7 ../data/Buzzard2015_data.csv | head -n 1

echo "Número de valores distintos"

cut -d ',' -f 7 ../data/Buzzard2015_data.csv | tail -n +2 | sort | uniq | wc -l

echo "Valor máximo"

cut -d ',' -f 7 ../data/Buzzard2015_data.csv | tail -n +2 | sort -n | tail -n 1

echo "Valor mínimo"

cut -d ',' -f 7 ../data/Buzzard2015_data.csv | tail -n +2 | sort -n | head -n 1
```