

Tarea 3: TC3_ForLoop

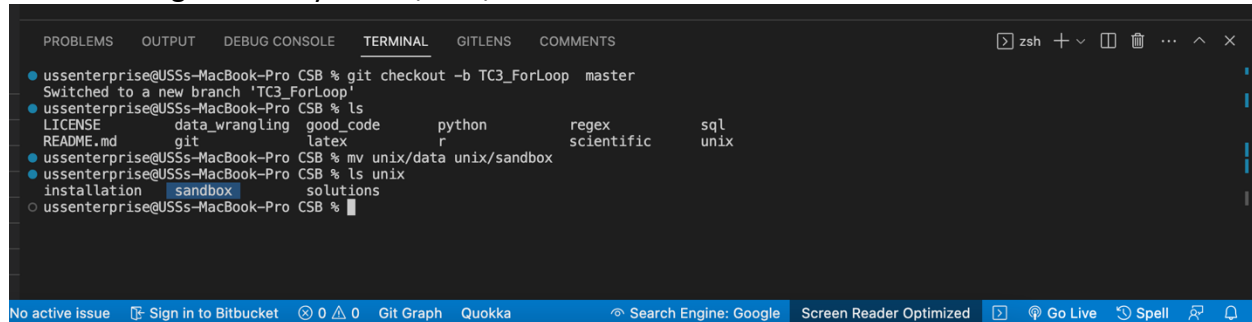
Nombre: Ronald Rivera

Curso: G02

Docente: Moisés Gualapuro

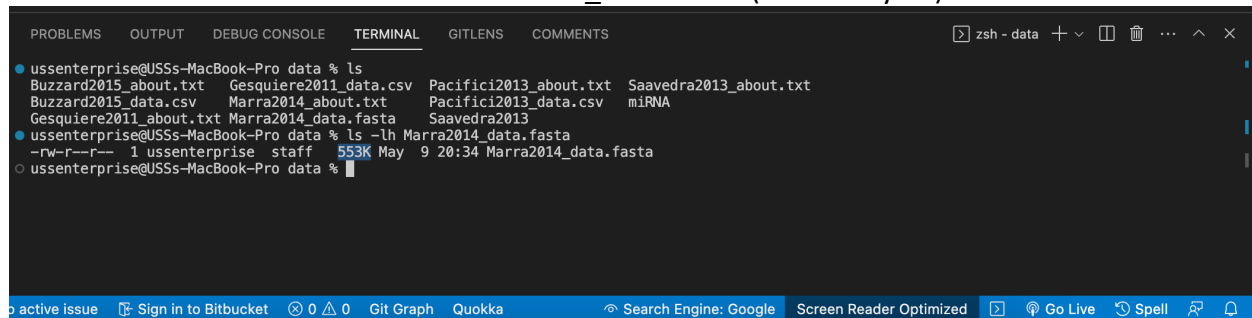
1.10.1 Next Generation Sequencing Data

1. Change directory to CSB/unix/sandbox.



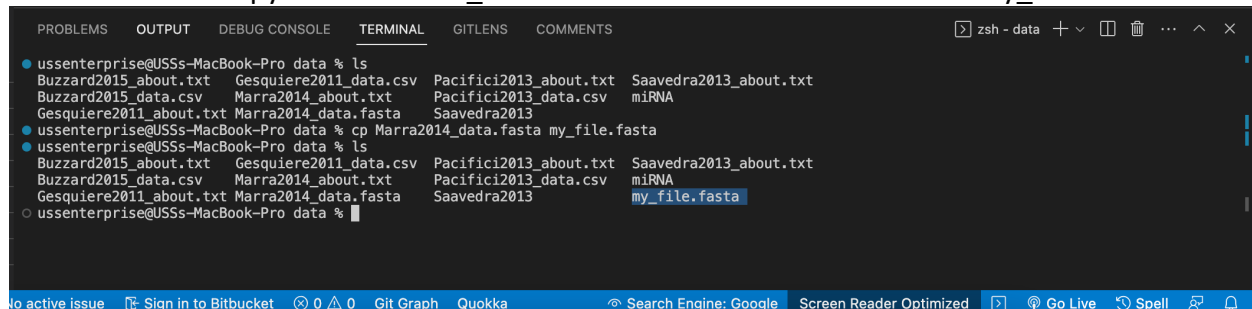
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS COMMENTS
ussenterprise@USSs-MacBook-Pro CSB % git checkout -b TC3_ForLoop master
Switched to a new branch 'TC3_ForLoop'
ussenterprise@USSs-MacBook-Pro CSB % ls
LICENSE      data_wrangling  good_code      python         regex          sql
README.md    git             latex          r              scientific
ussenterprise@USSs-MacBook-Pro CSB % mv unix/data unix/sandbox
ussenterprise@USSs-MacBook-Pro CSB % ls unix
installation  sandbox         solutions
ussenterprise@USSs-MacBook-Pro CSB %
```

2. What is the size of the file Marra2014_data.fasta? (553 Kilobytes)



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS COMMENTS
ussenterprise@USSs-MacBook-Pro data % ls
Buzzard2015_about.txt  Gesquiere2011_data.csv  Pacifici2013_about.txt  Saavedra2013_about.txt
Buzzard2015_data.csv   Marra2014_about.txt     Pacifici2013_data.csv   miRNA
Gesquiere2011_about.txt Marra2014_data.fasta    Saavedra2013
ussenterprise@USSs-MacBook-Pro data % ls -lh Marra2014_data.fasta
-rw-r--r-- 1 ussenterprise staff 553K May 9 20:34 Marra2014_data.fasta
ussenterprise@USSs-MacBook-Pro data %
```

3. Create a copy of Marra2014_data.fasta in the sandbox and name it my_file.fasta.



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS COMMENTS
ussenterprise@USSs-MacBook-Pro data % ls
Buzzard2015_about.txt  Gesquiere2011_data.csv  Pacifici2013_about.txt  Saavedra2013_about.txt
Buzzard2015_data.csv   Marra2014_about.txt     Pacifici2013_data.csv   miRNA
Gesquiere2011_about.txt Marra2014_data.fasta    Saavedra2013
ussenterprise@USSs-MacBook-Pro data % cp Marra2014_data.fasta my_file.fasta
ussenterprise@USSs-MacBook-Pro data % ls
Buzzard2015_about.txt  Gesquiere2011_data.csv  Pacifici2013_about.txt  Saavedra2013_about.txt
Buzzard2015_data.csv   Marra2014_about.txt     Pacifici2013_data.csv   miRNA
Gesquiere2011_about.txt Marra2014_data.fasta    Saavedra2013           my_file.fasta
ussenterprise@USSs-MacBook-Pro data %
```

4. How many contigs are classified as isogroup00036? (16 configs)

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS COMMENTS
ussenterprise@USSs-MacBook-Pro data % grep -c isogroup00036 my_file.fasta
16
ussenterprise@USSs-MacBook-Pro data %
```

5. Replace the original "two-spaces" delimiter with a comma. (los 2 espacios son remplazados por la coma en el cat y el head me permite ver las 3 lineas y validar si en comando funciona)

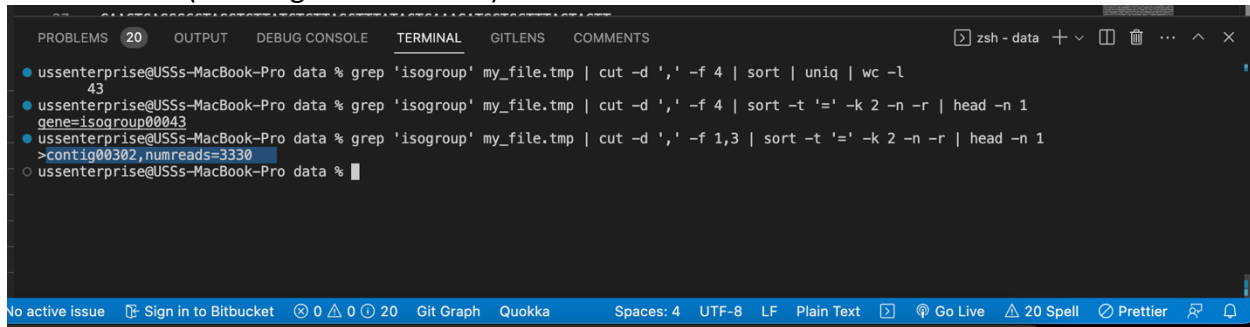
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS COMMENTS
ussenterprise@USSs-MacBook-Pro data % cat my_file.fasta | tr -s ' ', ' | head -n 3
>contig00001,length=527,numreads=2,gene=isogroup00001,status=it_thresh
ATCCTAGCTACTCTGGAGACTGAGGATTGAAGTTCAAAGTCAGCTCAAGCAAGAGATTG
TTTACAATTAACCCACAAAAGGCTGTTACTGAAGGTGTGGCTTAAGTGTGAGAGCAACAG
ussenterprise@USSs-MacBook-Pro data %
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS COMMENTS
ussenterprise@USSs-MacBook-Pro data % cat my_file.fasta | tr -s ' ', ' | head -n 3
>contig00001,length=527,numreads=2,gene=isogroup00001,status=it_thresh
ATCCTAGCTACTCTGGAGACTGAGGATTGAAGTTCAAAGTCAGCTCAAGCAAGAGATTG
TTTACAATTAACCCACAAAAGGCTGTTACTGAAGGTGTGGCTTAAGTGTGAGAGCAACAG
ussenterprise@USSs-MacBook-Pro data % cat my_file.fasta | tr -s ' ', ' > my_file.tmp
ussenterprise@USSs-MacBook-Pro data % ls
Buzzard2015_about.txt  Gesquiere2011_data.csv  Pacifici2013_about.txt  Saavedra2013_about.txt  my_file.tmp
Buzzard2015_data.csv  Marra2014_about.txt    Pacifici2013_data.csv  miRNA
Gesquiere2011_about.txt  Marra2014_data.fasta  Saavedra2013          my_file.fasta
ussenterprise@USSs-MacBook-Pro data % cat my_file.tmp | head -n 3
>contig00001,length=527,numreads=2,gene=isogroup00001,status=it_thresh
ATCCTAGCTACTCTGGAGACTGAGGATTGAAGTTCAAAGTCAGCTCAAGCAAGAGATTG
TTTACAATTAACCCACAAAAGGCTGTTACTGAAGGTGTGGCTTAAGTGTGAGAGCAACAG
ussenterprise@USSs-MacBook-Pro data %
```

6. How many unique isogroups are in the file? (43 grupos)

```
PROBLEMS 20 OUTPUT DEBUG CONSOLE TERMINAL GITLENS COMMENTS
ussenterprise@USSs-MacBook-Pro data % grep 'isogroup' my_file.tmp | cut -d ',' -f 4 | sort | uniq | wc -l
43
ussenterprise@USSs-MacBook-Pro data %
```

7. Which contig has the highest number of reads (numreads)? How many reads does it have? (la configuracion 00302)

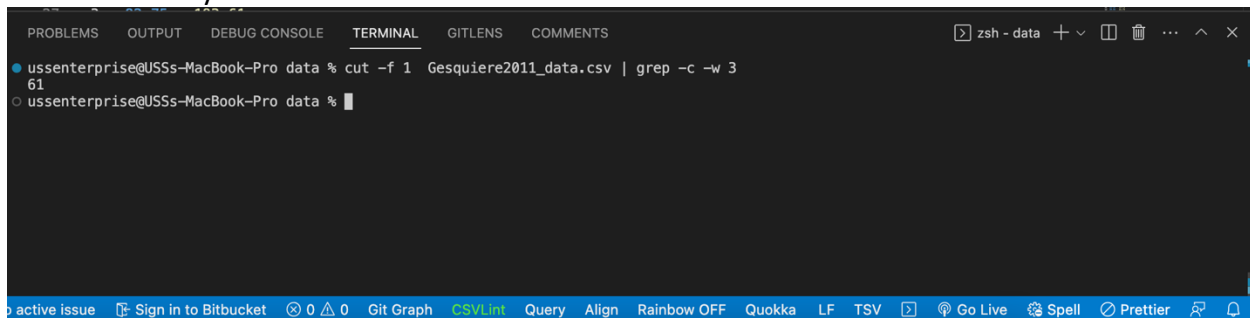


```
ussenterprise@USSs-MacBook-Pro data % grep 'isogroup' my_file.tmp | cut -d ',' -f 4 | sort | uniq | wc -l
43
ussenterprise@USSs-MacBook-Pro data % grep 'isogroup' my_file.tmp | cut -d ',' -f 4 | sort -t '=' -k 2 -n -r | head -n 1
gene=isogroup00043
ussenterprise@USSs-MacBook-Pro data % grep 'isogroup' my_file.tmp | cut -d ',' -f 1,3 | sort -t '=' -k 2 -n -r | head -n 1
>contig00302,numreads=3330
ussenterprise@USSs-MacBook-Pro data %
```

1.10.2 Hormone Levels in Baboons

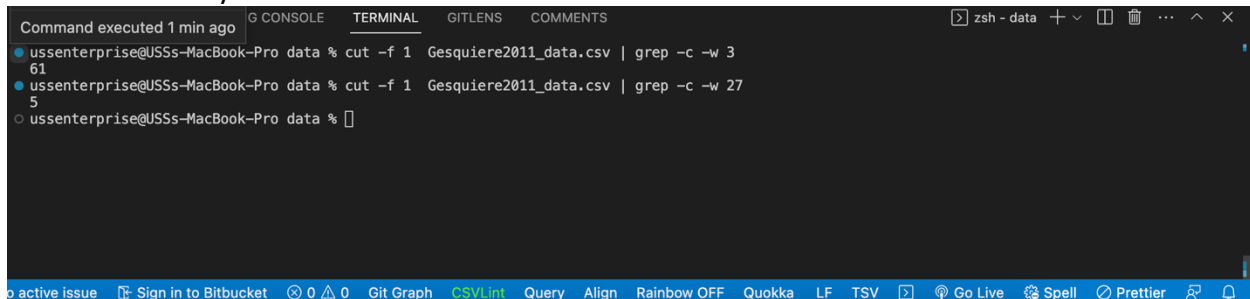
1. How many times were the levels of individuals 3 and 27 recorded?

En el nivel 3 hay 61



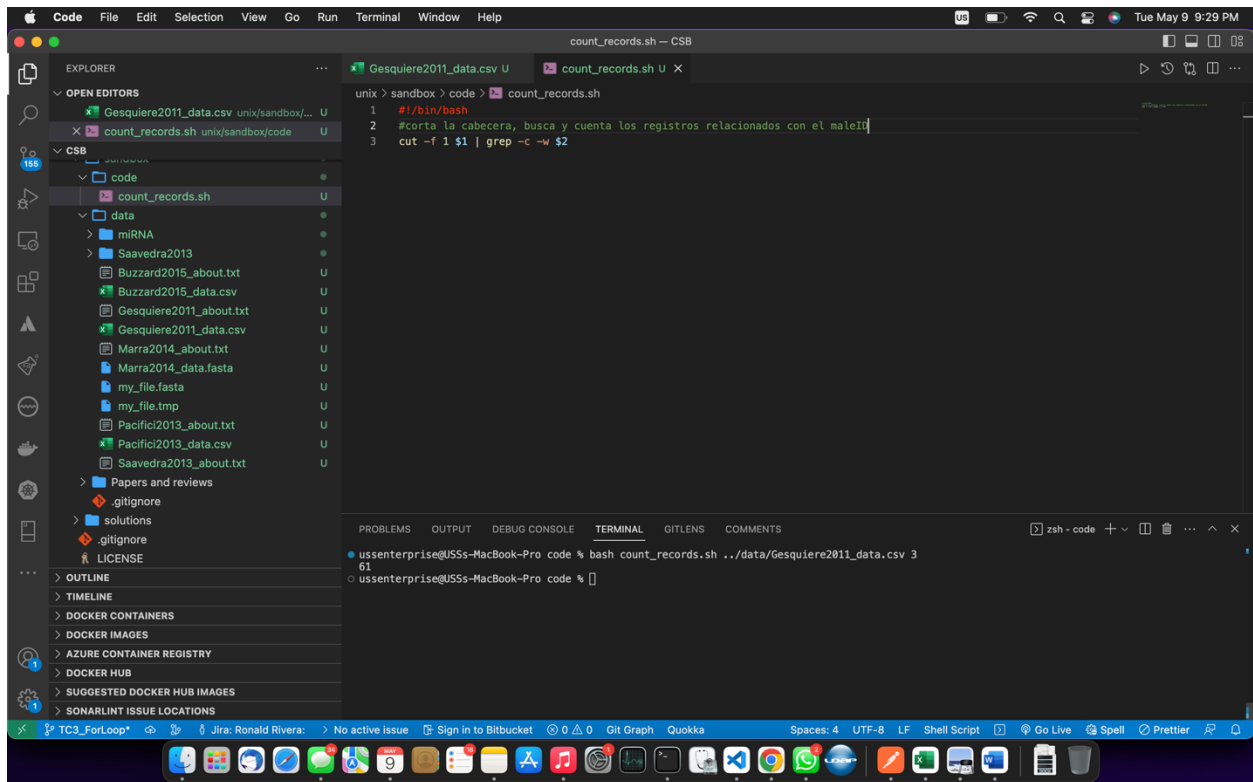
```
ussenterprise@USSs-MacBook-Pro data % cut -f 1 Gesquiere2011_data.csv | grep -c -w 3
61
ussenterprise@USSs-MacBook-Pro data %
```

En el nivel 27 hay 5



```
Command executed 1 min ago
ussenterprise@USSs-MacBook-Pro data % cut -f 1 Gesquiere2011_data.csv | grep -c -w 3
61
ussenterprise@USSs-MacBook-Pro data % cut -f 1 Gesquiere2011_data.csv | grep -c -w 27
5
ussenterprise@USSs-MacBook-Pro data %
```

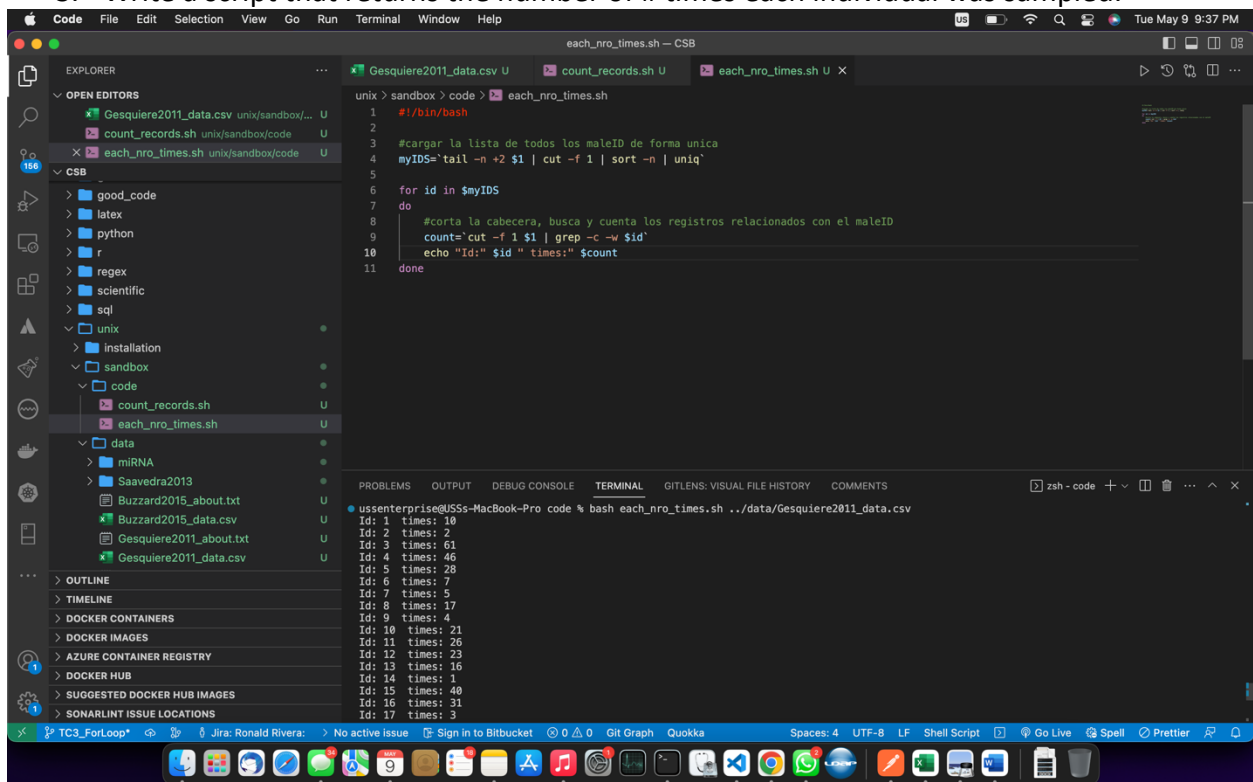
2. Write a script taking as input the file name and the ID of the individual, and returning the number of records for that ID.



```
count_records.sh — CSB
unix > sandbox > code > count_records.sh
1 #!/bin/bash
2 #corta la cabecera, busca y cuenta los registros relacionados con el maleID
3 cut -f 1 $1 | grep -c -w $2

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS COMMENTS
ussenterprise@USSs-MacBook-Pro code % bash count_records.sh ../data/Gesquiere2011_data.csv 3
ussenterprise@USSs-MacBook-Pro code %
```

3. Write a script that returns the number of # times each individual was sampled.

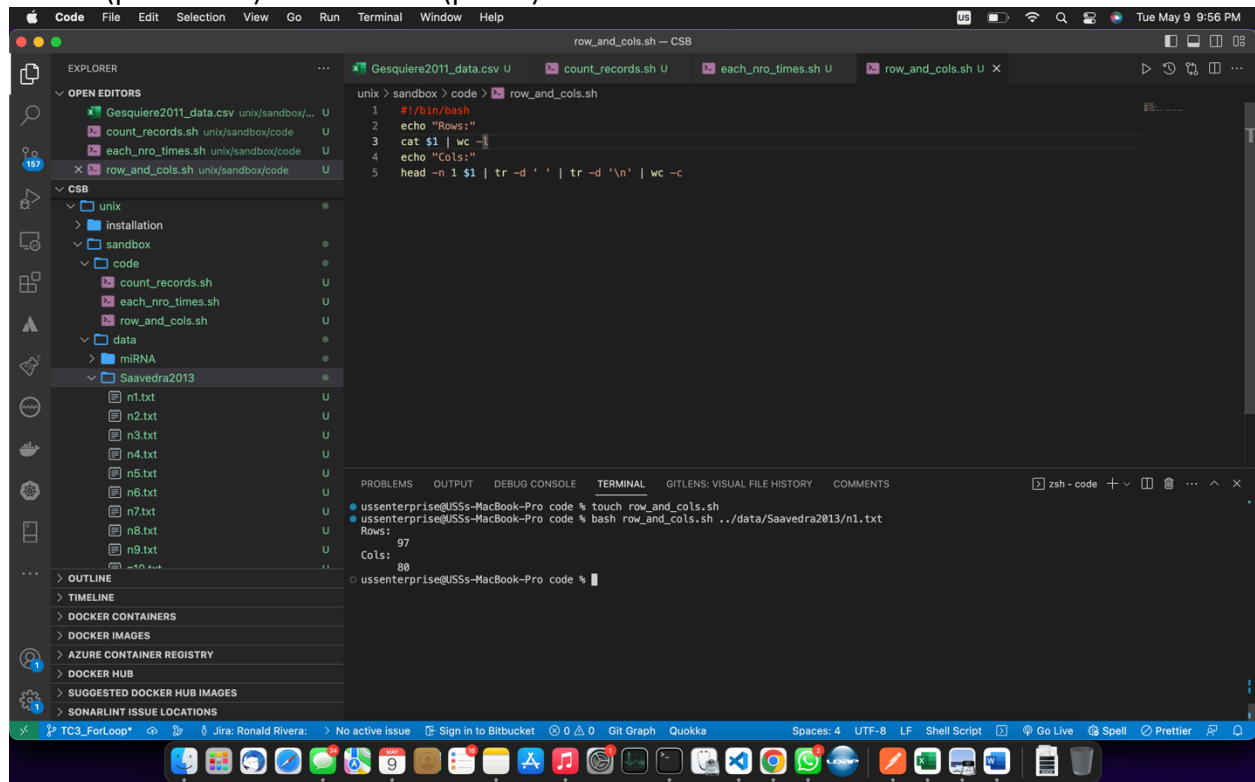


```
each_nro_times.sh — CSB
unix > sandbox > code > each_nro_times.sh
1 #!/bin/bash
2 #cargar la lista de todos los maleID de forma unica
3 myIDS='tail -n +2 $1 | cut -f 1 | sort -n | uniq'
4
5 for id in $myIDS
6 do
7
8     #corta la cabecera, busca y cuenta los registros relacionados con el maleID
9     count='cut -f 1 $1 | grep -c -w $id'
10    echo "Id:" $id " times:" $count
11 done

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS: VISUAL FILE HISTORY COMMENTS
ussenterprise@USSs-MacBook-Pro code % bash each_nro_times.sh ../data/Gesquiere2011_data.csv
Id: 1 times: 10
Id: 2 times: 2
Id: 3 times: 61
Id: 4 times: 46
Id: 5 times: 28
Id: 6 times: 7
Id: 7 times: 5
Id: 8 times: 17
Id: 9 times: 4
Id: 10 times: 21
Id: 11 times: 26
Id: 12 times: 23
Id: 13 times: 16
Id: 14 times: 1
Id: 15 times: 40
Id: 16 times: 31
Id: 17 times: 3
```

1.10.3 Plant–Pollinator Networks

1. Write a script that takes one of these files and determines the number of rows (pollinators) and columns (plants).



The screenshot shows a VS Code editor window with the following components:

- EXPLORER:** A file tree on the left showing a project structure with folders like 'unix', 'sandbox', 'code', 'data', 'miRNA', and 'Saavedra2013'. The file 'row_and_cols.sh' is selected under the 'code' folder.
- EDITOR:** The main workspace shows the content of 'row_and_cols.sh':

```
1 #!/bin/bash
2 echo "Rows:"
3 cat $1 | wc -l
4 echo "Cols:"
5 head -n 1 $1 | tr -d ' ' | tr -d '\n' | wc -c
```
- TERMINAL:** A terminal window at the bottom shows the execution of the script:

```
ussententerprise@USSs-MacBook-Pro code % touch row_and_cols.sh
ussententerprise@USSs-MacBook-Pro code % bash row_and_cols.sh ../data/Saavedra2013/n1.txt
Rows:
97
Cols:
80
```

2. Write a script that prints the number of rows and columns for each network

The screenshot shows a VS Code editor with a file explorer on the left and a terminal at the bottom. The file explorer shows a project structure with a 'data' directory containing a 'Saavedra2013' subdirectory with files n1.txt through n7.txt. The terminal shows the execution of a script 'row_and_cols_all.sh' which processes these files and outputs their row and column counts.

```
1 #!/bin/bash
2 #llama a todos los archivos .txt de la carpeta
3 FILES=../data/Saavedra2013/*.txt
4
5 for f in $FILES
6 do
7     rows=`cat $f | wc -l`
8     cols=`head -n 1 $f | tr -d ' ' | tr -d '\n' | wc -c`
9     echo "File: $f Rows: $rows Cols: $cols"
10 done
```

Terminal Output:

```
ussenterprise@USSs-MacBook-Pro code % bash row_and_cols_all.sh
File: ../data/Saavedra2013/n1.txt Rows: 37 Cols: 88
File: ../data/Saavedra2013/n10.txt Rows: 14 Cols: 20
File: ../data/Saavedra2013/n11.txt Rows: 270 Cols: 91
File: ../data/Saavedra2013/n12.txt Rows: 7 Cols: 72
File: ../data/Saavedra2013/n13.txt Rows: 61 Cols: 17
File: ../data/Saavedra2013/n14.txt Rows: 35 Cols: 15
File: ../data/Saavedra2013/n15.txt Rows: 38 Cols: 11
File: ../data/Saavedra2013/n16.txt Rows: 118 Cols: 24
File: ../data/Saavedra2013/n17.txt Rows: 76 Cols: 31
File: ../data/Saavedra2013/n18.txt Rows: 13 Cols: 14
File: ../data/Saavedra2013/n19.txt Rows: 10 Cols: 16
File: ../data/Saavedra2013/n2.txt Rows: 62 Cols: 41
File: ../data/Saavedra2013/n20.txt Rows: 18 Cols: 7
File: ../data/Saavedra2013/n21.txt Rows: 19 Cols: 45
File: ../data/Saavedra2013/n22.txt Rows: 19 Cols: 36
File: ../data/Saavedra2013/n23.txt Rows: 179 Cols: 76
File: ../data/Saavedra2013/n24.txt Rows: 80 Cols: 76
```

3. Which is the network with the largest number of rows? Which the one with the largest number of columns?

El número más grande de las columnas (207 n56.txt)

The screenshot shows the same VS Code editor setup, but the terminal now shows a command to sort the files by column count and display the top 6 results.

```
ussenterprise@USSs-MacBook-Pro code % bash row_and_cols_all.sh | sort -n -r -k 6 | head -n 1
File: ../data/Saavedra2013/n56.txt Rows: 110 Cols: 207
```

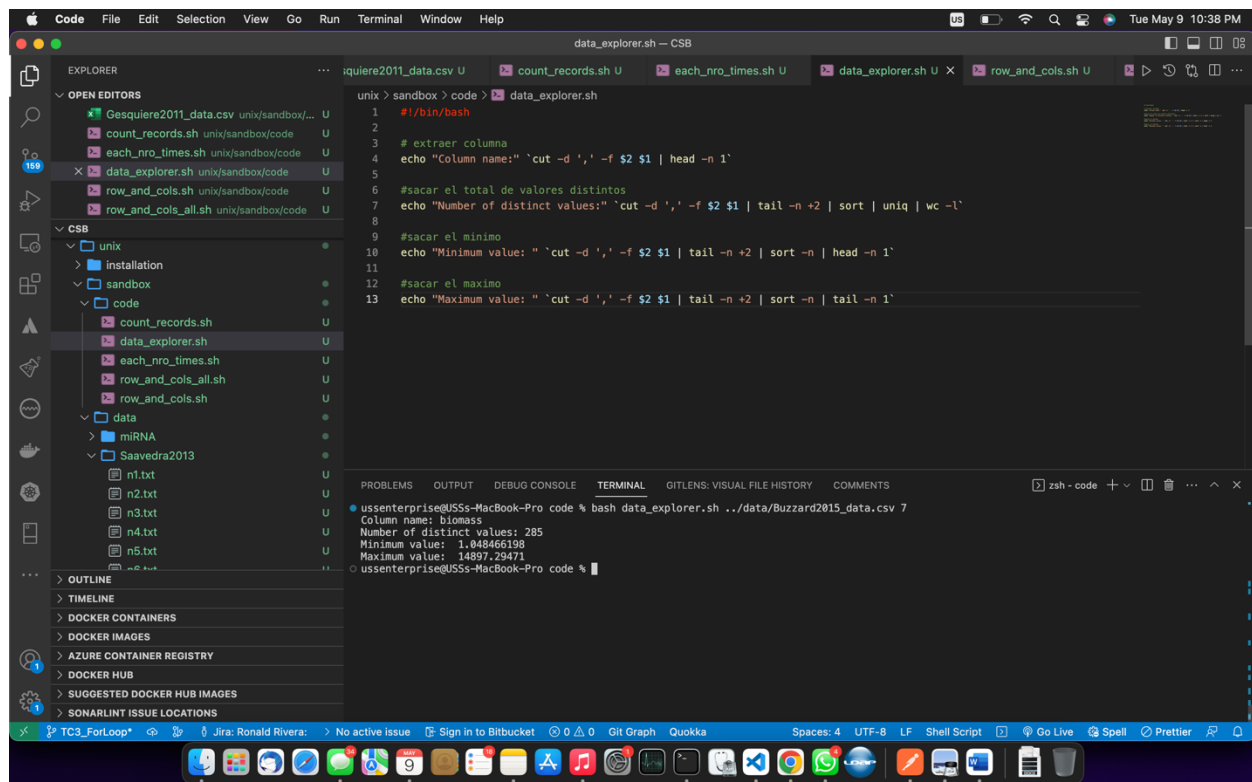
El número más grande de las filas (678 n58.txt)

```
row_and_cols_all.sh — CSB
unix > sandbox > code > row_and_cols_all.sh
1 #!/bin/bash
2 #llama a todos los archivos .txt de la carpeta
3 FILES=../data/Saavedra2013/*.txt
4
5 for f in $FILES
6 do
7     rows=$(cat $f | wc -l)
8     cols=$(head -n 1 $f | tr -d ' ' | tr -d '\n' | wc -c)
9     echo "File: $f Rows: $rows Cols: $cols"
10 done
```

```
ussenterpriseUSSs-MacBook-Pro code % bash row_and_cols_all.sh | sort -n -r -k 4 | head -n 1
File: ../data/Saavedra2013/n58.txt Rows: 678 Cols: 90
ussenterpriseUSSs-MacBook-Pro code %
```

1.10.4 Data Explorer

1. Write a script that, for a given csv file and column number, prints:
 - Column name
 - Number of distinct values
 - Minimum value
 - Maximum value



Scripts subidos:

https://github.com/ronaldsoft/CSB/tree/TC3_ForLoop/unix/sandbox/code

Rama: TC3_ForLoop