

Introducción a Linux (usando Bash)

✉ consultas: gabriel.lichtenstein@gmail.com

Si aún no lo hicieron, pueden ver la guía para conectarse a nuestro clúster aquí: [¿Cómo conectarse al clúster para realizar los trabajos prácticos?](#)

Introducción

Ahora sí, una vez conectados al clúster **bioinfo.fmed.uba.ar**, usaremos una Terminal para interactuar con el mismo, esta interfaz es un intérprete de comandos de texto del tipo BASH ([Bourne Again Shell](#)).

En otras palabras, una terminal bash es una interfaz que espera comandos de texto a ser escritos por el usuario mediante el teclado, cuando se ejecutan los comandos escritos estos son enviados a la computadora en lenguaje binario, que es el lenguaje que la computadora comprende, es por eso que a la terminal también se la conoce como intérprete de comandos. Nosotros escribimos dentro de la terminal en lenguaje Bash y el intérprete traduce nuestros comandos y se los envía a la computadora para ser ejecutados. Dentro de la terminal el mouse no se utiliza para ejecutar comandos o programas, solo sirve, por ejemplo, para seleccionar texto, copiarlo y pegarlo o para mover la barra lateral para subir y bajar la página en la que estamos. Sin embargo, para navegar el sistema de archivos, abrir carpetas y ejecutar programas los comandos deben ser escritos de manera literal en el intérprete. Cada vez que escribimos un comando, lo ejecutamos presionando la tecla Enter↵. Hay otros lenguajes además de Bash, cada uno con su objetivo puntual, con sus pros y cons, por ejemplo: python, perl, java y R. Pero para la mayoría de las interacciones con el clúster y su sistema operativo, usaremos Bash.

BASH

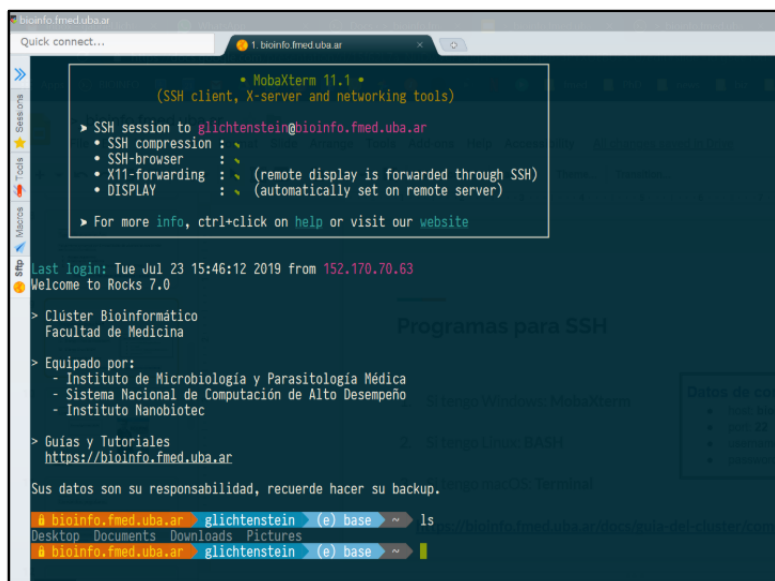
Bash es el intérprete de comandos que usaremos al conectarnos a bioinfo.

Bash es la interfaz para manejar nuestros datos y ejecutar programas y *scripts*.

La interfaz Bash se maneja a través del teclado, no requiere de un *mouse*.

Bash se usa mucho para trabajar con grandes archivos de texto plano (e.j: *fastas*).

Bash es el lenguaje que usaremos para escribir nuestros *scripts*.



Bash contiene decenas de comandos que pueden escribirse para ejecutarse en la terminal y junto con algunas convenciones de semántica y sintaxis componen un robusto lenguaje de programación enfocado al [scripting](#) con el cual pueden implementarse complejos algoritmos. En este tutorial solamente veremos algunos [comandos populares](#), como para poder comenzar a usarlo inmediatamente e interactuar con el servidor y nuestros datos.

Comando	Acción	Comando	Acción
<i>ls</i>	Listar contenidos	<i>head</i>	Ver 10 primeras líneas
<i>cd</i>	Change directorio	<i>tail</i>	Ver 10 últimas líneas
<i>mkdir</i>	Make directorio	<i>nano</i> ó <i>vim</i>	editores de texto
<i>pwd</i>	Path to Working Directory	<i>cp</i>	Copiar
<i>cat</i>	Concatenate & show	<i>mv</i>	Mover o renombrar
<i>more</i>	Muestra página por página		Un <i>pipe</i> , para <i>pipelines</i>
<i>grep</i>	Buscar <i>strings</i> en textos	<i>wc</i>	Word Count
<i>wget</i>	Descarga de la web	<i>gunzip</i>	Descomprime archivos .gz

La mejor manera de aprender estos comandos es usándolos para ver qué hacen. En la siguiente sección realizaremos algunas tareas y una evaluación para familiarizarnos con ellos.

Manos a la obra...

Tareas

Una vez que ingresó al clúster debe leer la tabla de comandos populares de Bash para completar las tareas y la evaluación. Dentro de cada tarea (1-20) he puesto algunas pistas (Tips) para que logren completarlas, son solo pistas por lo que deben ajustarlas un poco para poder cumplir con la tarea. El ejercicio es sencillo, es para que puedan familiarizarse con algunos comandos. Algo que todos los comandos van a tener en común, es que pueden leerse sus manuales y/o también una hoja de ayuda más resumida. Por ejemplo, si escribo y ejecuto el comando **ls**, este simplemente hará su tarea predeterminada (*default*), pero si quiero ver qué más puede hacer el programa, o simplemente como se usa, podemos pasarle el parámetro **--help**, es decir, escribo y ejecuto: **ls --help**, esto imprimirá una hoja de ayuda. Este parámetro **--help** o a veces simplemente **-h**, puede usarse para la mayoría de los comandos de Bash. Una alternativa es usar un comando para ver el manual, este comando es simplemente: **man**. Si quiero ver el manual de un comando, puedo hacer: **man ls**, y esto puedo usarlo para cualquier otro comando como: **man grep**, etc.

Nota: Los comandos de bash están escritos con fondo gris

1-Crear la carpeta **cluster101** dentro de su [carpeta de usuario \(Home\)](#)

Tip: **mkdir nombredemicarpeta**

2-Crear la carpeta **cluster102** dentro de su Home

3-Borre la carpeta **cluster102**

Tip1: **rm nombredearchivo**

Tip2: **rm -r nombredecarpeta**

Es buen momento para recordarles que en bash no hay opción deshacer, es decir, una vez que borramos la carpeta **cluster102** esta no puede ser recuperada, no hay papelera de reciclaje en la terminal.

4- Ingresar a la carpeta cluster101

Tip: `cd` carpeta

5-Descargar desde internet el archivo FASTQ: SP1.fq

No le haga click, usando `wget` descargue el archivo fastq con este link:

https://molb7621.github.io/workshop/_downloads/SP1.fq

Tip: `wget` URL

Tip2: URL es el acrónimo usado para decir: dirección de internet, en este caso donde dice URL debe usted escribir <https://mol...fq>

Ej: `wget https://mol...fq`

Para ver si se descargo el archivo, liste el contenido de su carpeta **cluster101**.

Tip: `ls`

Tip2: `ls *`

Tip3: `ls -hal`

Tip4: `ls -hal *.fq`

Con el tip3 y/o 4 se puede ver en la 5ta columna el tamaño de los archivos (kilobytes, megabytes, etc), en las demás columnas se ven detalles como fechas y permisos.

6-Descargar usando wget el archivo FASTA: sample.fa

https://molb7621.github.io/workshop/_downloads/sample.fa

7-Descargar usando wget el archivo FASTA de proteínas que está comprimido con el formato **gunzip**: UP000004994_4081.fasta.gz

ftp://ftp.uniprot.org/pub/databases/uniprot/current_release/knowledgebase/reference_proteomes/Eukaryota/UP000004994_4081.fasta.gz

8-Descomprimir el archivo FASTA de proteínas

Antes de descomprimir el archivo puede hacer `ls -hal` para ver cuanto pesa el archivo en Megabytes que acaba de bajar.

Tip: `gunzip miarchivo.gz`

Luego de descomprimir el archivo puede hacer `ls -hal` para ver cuanto pesa en Megabytes el archivo que acaba de descomprimir. Verá que un fasta descomprimido ocupa más del doble que un fasta comprimido. Por eso estos archivos suelen estar comprimidos al descargarlos.

9-Explorar el contenido de los archivos descargados

Tip1: `cat archivo.fa`

Tip2: `cat archivo.fasta`

Tip3: `cat archivo.fq`

Tip4: `head archivo.##`

Tip5: `tail archivo.##`

Tip6: `more archivo.##`

Usando `more`, se puede pasar de página usando la barra espaciadora y para salir se puede hacer cancelar apretando la combinación de teclas: `Ctrl c`

Pro-tip: usando la tecla `Tab` se pueden autocompletar comandos y nombres de archivos y carpetas. Por ejemplo si escribo `tail archi` y luego presiono una vez la tecla `Tab` se autocompletara el resto del nombre del archivo: `tail archivo.fasta` - Si por ejemplo escribo `mo` y presiono dos veces la tecla `Tab` se mostrará un listado de comandos que comienzan con las letras `mo`, por ejemplo: `more`

10-Crear 3 carpetas dentro de `cluster101`:

Carpeta 1) `fasta`

Tip: `mkdir fasta`

Carpeta 2) `fastq`

Carpeta 3) `resultados`

11-Mover los archivos descargados a sus carpetas `fasta` y `fastq` correspondientes.

Tip1: Los archivos `fasta` pueden tener las siguientes extensiones:
`.fasta`; `.fa`; `.fna` (nucleótidos) o `.faa` (aminoácidos), etc.

Tip2: Los archivos `fastq` suelen ser:
`.fastq` o `.fq`.

Tip3: `mv archivo carpetadedestino/`

Tip4: `mv *.fasta fasta/`

12- Cuente cuantas líneas tiene el archivo `sample.fa`

Tip1: `cat archivo.fa | wc`

Tip2: Recuerde que el archivo ahora está en una carpeta, así que puede ingresar a la carpeta con `cd fasta`, o puede ejecutar `cat fasta/archivo.fa`

Tip3: `cat archivo.fa | wc -l`

Recuerde que queremos contar solo líneas, no palabras, ni caracteres.

13- Cuente cuantas secuencias tiene el archivo `sample.fa`

Tip1: En un archivo `fasta`, cada secuencia comienza con el símbolo `>`

Tip2: `grep ">" archivo.fa`

Tip3: `grep ">" archivo.fa | wc -l`

Tip4: `grep -c ">" archivo.fa`

El Tip3 y el 4, darán el mismo resultado, el Tip3 usa la forma de pipeline, esta forma se puede usar para muchas otras cosas, como `cat archivo.fa | wc -l`, pero para contar las ocurrencias del resultado de `grep` hay un parámetro `-c` para contar y que aprovechamos usando el Tip4.

14-Posiciónese en la carpeta fastq y descargue el archivo *RPPK30_1.fastq.gz*

https://sra-pub-src-2.s3.amazonaws.com/SRR10127552/RPPK30_1.fastq.gz.1

Tip1: Supongamos que usted ingresó previamente a la carpeta **fasta** escribiendo: `cd fasta`, para volver a la carpeta donde estaba antes (es decir: **cluster101**) se puede usar este comando: `cd ..` para retroceder a la carpeta donde estaba antes. También puede escribir solamente `cd`, con lo cual regresará a su carpeta home y luego puede escribir la ruta completa: `cd cluster101/carpetadestino`

Tip2: `cd ..`

Tip3: `cd carpetadestino/`

Tip4: `wget URL`

Tip5: `ls -hal`

15-Cambie el nombre del archivo *RPPK30_1.fastq.gz.1* por el nombre *RPPK30_1.fastq.gz*

Tip: `mv nombredearchivo nuevonombredearchivo`

16-Descomprima el archivo *RPPK30_1.fastq.gz***17- Posiciónese en la carpeta resultados****18- Dentro de la carpeta resultados cree un archivo de texto llamado *saludo.sh***

Tip1: Recuerde el pto. 14

Tip2: `nano saludo.sh`

Escriba dentro del archivo:

```
echo "Hola Nombre y Apellido"
```

Tip3: Ctrl x es la combinación de teclas usada para salir/**exit**, pero antes le preguntará si va a guarda el archivo.

Pro-tip: En nano el símbolo ^ representa la tecla Ctrl.

19- Ejecute el script de bash *saludo.sh*

Tip: `bash saludo.sh`

20- Vuelva a posicionarse en su carpeta home

Tip1: `pwd`

Tip2: `cd`

Tip3: `pwd`

Evaluación

Responda las siguientes preguntas y guarde el resultado en un archivo de texto llamado `tu_usuario.txt` (reemplace "tu_usuario" por su nombre de usuario) dentro de la carpeta **resultados**.

- 1) ¿Cuántas secuencias hay en cada uno de los archivos fasta?
- 2) ¿Cómo se llama la última secuencia del archivo UP000004994_4081.fasta?
- 3) ¿Cuántas secuencias hay en el archivo RPPK30_1.fastq?

Tip: Al hacer head y luego tail sobre nuestro archivo fastq, vemos que la primer secuencia y la última secuencia comienzan de la misma manera. Es decir comparten la primer parte del nombre. Por ejemplo se verá algo como @M4555: entonces podemos hacer un grep y contar cuántas veces aparece esta expresión en el archivo fastq y esto nos dará el número de secuencias. Otra opción es contar el número de líneas y [dividir el resultado por 4](#).

- 4) ¿Porque cree que no contamos solamente las apariciones del caracter "@" como cuando contamos las apariciones del caracter ">" para el archivo fasta?
- 5) ¿Qué tamaño en Megabytes tiene el archivo RPPK30_1.fastq?

¡Gane puntos extra y una medalla!

Realice un backup de su carpeta **cluster101** en su computadora personal.

Tip: Mediante la solapa **sftp** del **mobaXterm** podrá **subir** ↑ y **descargar** ↓ archivos y carpetas desde su computadora al servidor y viceversa.

Tip2: Utilizando el software [FileZilla](#) podrá hacer lo mismo pero de manera mucho más eficiente, sobre todo si sus archivos son grandes.

Si llego hasta aquí, completo las tareas, la evaluación y realizó su backup, usted se ha ganado esta magnífica medalla digital:





Except where otherwise noted, content on this site is licensed under a **Creative Commons Attribution 3.0 License**.