

### Taller Latinoamericano de Evolución Molecular 2ª edición, 17-28 Enero 2011

#### Introducción al biocómputo en sistemas UNIX/Linux

Pablo Vinuesa ([vinuesa@ccg.unam.mx](mailto:vinuesa@ccg.unam.mx))

Centro de Ciencias Genómicas-UNAM, México <http://www.ccg.unam.mx/~vinuesa/>

Todo el material del curso (presentaciones, lecturas, ejercicios, tutoriales, URLs ...) lo encontrarás en: <http://www.ccg.unam.mx/~vinuesa/tlem/>

#### Introducción al biocómputo en sistemas Unix/Linux

1. Qué es UNIX y Linux
2. Tengo una PC que corre windows, ¿cómo puedo correr Linux en mi máquina?
3. ¿Cómo hago trabajar a UNIX/Linux? -el shell: comandos y conceptos básicos
4. Uso de tuberías de comandos UNIX/Linux para procesar archivos de secuencias

### Introducción al biocómputo en sistemas UNIX/Linux

¿Qué es UNIX? - fuente: Wikipedia

Unix (registrado oficialmente como UNIX®) es un [sistema operativo portable, multitarea y multiusuario](#); desarrollado, en principio, en 1969 por un grupo de empleados de los [laboratorios Bell](#) de AT&T, entre los que figuran [Ken Thompson](#), [Dennis Ritchie](#) y [Douglas McIlroy](#).

UNIX es un Sistema Operativo no libre muy popular, porque está basado en una arquitectura que ha demostrado ser técnicamente estable.



Ken Thompson y Dennis Ritchie Fuente: Wikipedia

### Introducción al biocómputo en sistemas UNIX/Linux

La evolución de las familias UNIX - fuente: Wikipedia

Existen varias familias del sistema operativo UNIX que han evolucionado de manera independiente a lo largo de los años. Cada familia se distingue no tanto por sus diferencias técnicas como por sus diferencias en propiedad intelectual. Se observa que todas las familias se han visto contaminadas, directa o indirectamente, por otras familias. Las familias UNIX más significativas son:

**AT&T:** la familia que tuvo su origen en el UNIX de AT&T. Considerada la familia UNIX "pura" y original. Sus sistemas operativos más significativos son UNIX System III y UNIX System V.

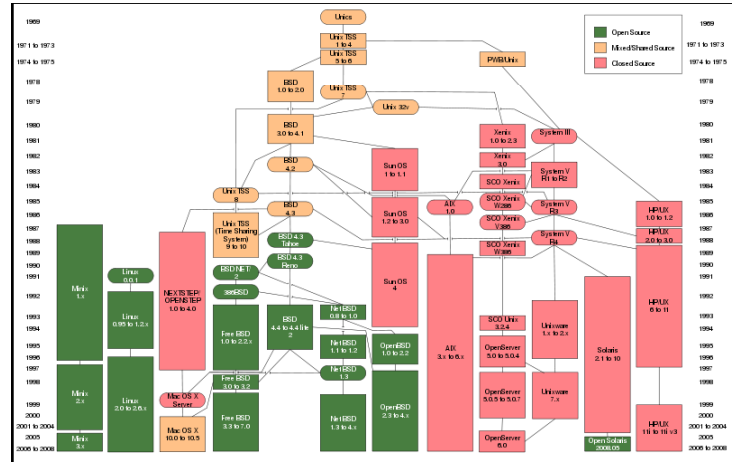
**BSD:** familia originada por el licenciamiento de UNIX a Berkely. BSD se reescribió para no incorporar propiedad intelectual originaria de AT&T en la versión 4. La primera implementación de los protocolos TCP/IP que dieron origen a Internet son la pila (stack) TCP/IP BSD.

**AIX:** esta familia surge por el licenciamiento de UNIX System III a IBM.

**Xenix:** familia derivada de la adquisición de los derechos originales de AT&T primero por parte de Microsoft y de esta los vendió a SCO.

### Introducción al biocómputo en sistemas UNIX/Linux

La evolución de las familias UNIX - Fuente Wikipedia



### Introducción al biocómputo en sistemas UNIX/Linux

- La evolución de las familias UNIX - fuente: Wikipedia y Sun Microsystems



<http://www.sun.com/software/solaris/>  
<http://learning-solaris.com/>  
<http://www.opensolaris.org/os/>  
<http://www.openoffice.org/>

Solaris es un sistema operativo de tipo Unix desarrollado por Sun Microsystems desde 1992 como sucesor de SunOS. Es un sistema certificado oficialmente como versión de Unix.

Funciona en arquitecturas SPARC y x86 para servidores y estaciones de trabajo. Aunque Solaris fue desarrollado como software privativo, la mayor parte de su código se ha liberado como proyecto de software libre denominado OpenSolaris. Solaris es conocido por su escalabilidad, especialmente en sistemas SPARC, y por ser origen de innovadoras tecnologías, como DTrace y ZFS.

El primer sistema operativo de Sun nació en 1983 y se llamó inicialmente SunOS. Estaba basado en el sistema UNIX BSD, de la Universidad de Berkeley. Más adelante incorporó funcionalidades del System V, convirtiéndose prácticamente en un sistema operativo totalmente basado en System V.

### Introducción al biocómputo en sistemas UNIX/Linux

- ¿Qué es Linux? - Evolución de sistemas UNIX y similares a UNIX (fuente: Wikipedia)

GNU: En 1983, Richard Stallman anunció el Proyecto GNU, un ambicioso esfuerzo para crear un sistema similar a Unix, que pudiese ser distribuido libremente. El software desarrollado por este proyecto -por ejemplo, GNU Emacs y GCC - también han sido parte fundamental de otros sistemas UNIX.



Richard Stallman



POWERED BY  
**Linux**



Linus Torvalds

Linux: En 1991, cuando Linus Torvalds empezó a proponer el núcleo Linux y a reunir colaboradores, las herramientas GNU eran la elección perfecta. Al combinarse ambos elementos, conformaron la base del sistema operativo (basado en POSIX) que hoy se conoce como GNU/Linux. Las distribuciones basadas en el núcleo, el software GNU y otros agregados entre las que se pueden mencionar a Red Hat Linux y Debian GNU/Linux se han hecho populares tanto entre los aficionados a la computación como en el mundo empresarial. Obsérvese que Linux tiene un origen independiente, por lo que se considera un 'clónico' de UNIX y no un UNIX en el sentido histórico.

### Introducción al biocómputo en sistemas UNIX/Linux

- Tengo una PC que corre windows, cómo puedo correr Linux en mi máquina?

#### 1. Puedes instalar Linux en una nueva partición.

descargas gratuitas de distribuciones desde:  
**Ubuntu** - <http://www.ubuntu.com/getubuntu/download>  
**OpenSuse** - <http://software.opensuse.org/>  
**Fedora** - <http://fedoraproject.org/es/get-fedora>

#### 2. Puedes instalar cygwin sin necesidad de particionar el disco.

<http://www.cygwin.com/>

#### 3. Puedes descargar Knoppix "liveCD" que no requiere instalación en el disco duro

<http://www.knopper.net/knoppix/index-en.html>  
<http://bioknoppix.hpcf.upr.edu/>

### Introducción al biocómputo en sistemas UNIX/Linux

- ¿cómo me conecto al servidor UNIX?

login: **quest??** (ej: **quest40**)  
 Password: **\*\*\*\*\***

Cambia tu password con: **passwd quest??**

Leyva Karla Yeriana	quest01
Sortibrán Andres	quest02
Vilchis Claudia Andrea	quest03
Isa Pavel	quest04
Pech Angelica	quest05
SIFUENTES ANA MARIA	quest06
PAJARES SILVIA	quest07
Torres Gonzalo	quest08
Blanco Juan	quest09
Lara Paloma	quest10
Martínez Mario Alberto	quest11
GUEVARA LAZARO	quest12
González Andrea	quest13
Aguilar Gisela	quest14
Montalvo Javier	quest15
Gómez Zulema	quest16
Ortiz Daniel Dario	quest17
Suárez David	quest18
Lozada Alejandro Nabor	quest19
Pasos Silvia	quest20

Vázquez Monserrat	quest21
Becerra Itzel	quest22
Morales Ariadna Esthela	quest23
Cruz Pablo	quest24
Avitia Morena	quest25
Martinez Fernando	quest26
GARCIA FRANCISCO JAVIER	quest27
Riquelme Cristina	quest28
Rodríguez Hernando	quest29
Godoy Elizabeth Ernestina	quest30
Santillán Veronica	quest31
Olvera Aurora	quest32
Salvador Miguel	quest33
SERRANO ANGELICA	quest34
Serrano Victor Manuel	quest35
Torres Alejandro	quest36
RUIZ ANGELICA	quest37
Estrada Daniel A.	quest38
Sorbitan Rocio Luguí	quest39
Perez Carolina	quest40

Introducción al biocómputo en sistemas UNIX/Linux

¿puedo conectarme al servidor UNIX (xibalba) desde mi laptop? Si, definitivamente: usa la red inalámbrica con el passwd que verás en el pizarrón y Te logeas al sistema usando tu terminal (UNIX/Linux) o putty en windows, como se mostrará seguidamente

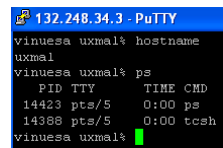
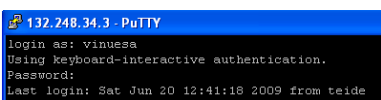
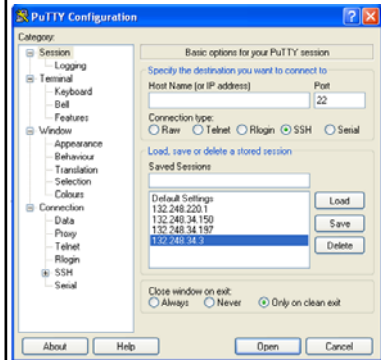
```
pablo@Tenerife:~$ ssh -x guest40@xibalba.lcg.unam.mx
Password:
Last login: Sun Jan 16 12:28:30 2011 from 189.215.23.180.
Sun Microsystems Inc. SunOS 5.10 Generic January 2005
You have mail.
-bash-3.00$
-bash-3.00$ hostname
Xibalba
-bash-3.00$ pwd
/home/guest40
-bash-3.00$ ls -a
. . . .bash_aliases .bash_history .bashrc .profile .staroffice8
-bash-3.00$ mkdir practicas_UNIX
-bash-3.00$ ls
practicas_UNIX
-bash-3.00$ exit
```

Introducción al biocómputo en sistemas UNIX/Linux

Tengo una laptop que corre windows, ¿cómo puedo acceder a xibalba desde mi máquina?

1. Instala PuTTY en tu máquina:

<http://chiark.greenend.org.uk/~sgtatham/putty/download.html>



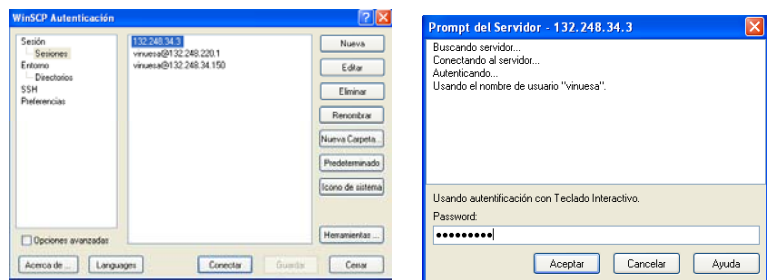
PuTTY, abre una consola en windows para establecer comunicación por ssh (Secure Shell) con un servidor UNIX

Introducción al biocómputo en sistemas UNIX/Linux

Tengo una PC que corre windows, ¿cómo puedo correr Linux en mi máquina?

Winscp abre una consola en windows para intercambiar archivos con servidores UNIX/Linux a través de un ambiente gráfico y scp.

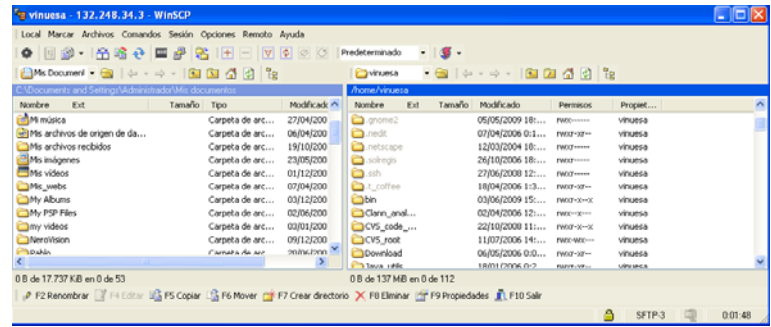
<http://sourceforge.net/projects/winscp/>



Introducción al biocómputo en sistemas UNIX/Linux

Tengo una PC que corre windows, ¿cómo puedo correr Linux en mi máquina?

Winscp abre una consola en windows para intercambiar archivos con servidores UNIX/Linux a través de un ambiente gráfico mediante scp



Ya puedes transferir archivos entre máquinas y directorios (carpetas) usando el protocolo secure copy (scp).

### Introducción al biocómputo en sistemas UNIX/Linux

- Ok, ya estoy conectado a una máquina UNIX/Linux ... ¿qué hago ahora?
- explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos
  - ¿Cómo hago trabajar a UNIX/Linux? - interpretación de comandos por el shell
- Una de las características más sobresalientes de estos sistemas operativos es la amplia gama de comandos (herramientas) con las que cuentan. **Más de 200 comandos** se distribuyen con el sistema operativo estándar. Cada una está especializada en hacer unas pocas cosas, pero muy bien, generalmente con múltiples opciones.
- El verdadero poder de estos sistemas operativos, a diferencia de Windows, radica en la elegancia y sencillez con la que **estos comandos pueden ser combinados** para realizar funciones mucho más sofisticadas que las que hace un comando individual.
- De manera consistente con esta filosofía, y para proveer de un ambiente de trabajo consistente y amigable para el usuario, los sistemas UNIX/Linux cuentan con la **consola de comandos o shell**. El shell es simplemente un programa que lee los comandos tecleados por el usuario en la línea de comandos (o guardados en un archivo, es decir, un programa o secuencia de comandos) convirtiéndolos en el lenguaje que entiende "las tripas" de la máquina. Para maximizar la eficiencia del sistema, el shell incluye construcciones de programación que permiten al usuario tomar decisiones, construir bucles y expresiones regulares, guardar valores en variables y leer y escribir texto, entre otras.

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos
- ¿Dónde encuentro una lista y descripción básica de los comandos disponibles?
 

Mira esta List of UNIX utilities de la Wikipedia:  
[http://en.wikipedia.org/wiki/List\\_of\\_Unix\\_programs](http://en.wikipedia.org/wiki/List_of_Unix_programs)
- ¿Qué es el shell?
 

Los sistemas UNIX/Linux se dividen lógicamente en dos piezas: el kernel y las utilidades.

El kernel es el corazón del sistema y reside en la memoria de la computadora desde el momento que se arranca y hasta que se apaga.

Las utilidades (comandos), residen en el disco físico y se cargan en memoria sólo cuando son llamadas.

El shell también es un programa. Se carga automáticamente en memoria desde que uno hace el log in a una máquina para que el usuario pueda interactuar con ella.

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos
- El login shell
 

Una vez dado el password, login verifica el nombre de usuario y la contraseña contra las entradas correspondientes en el archivo **/etc/passwd**.

Este archivo contiene una línea por cada usuario dado de alta en el sistema y la información mostrada:

```
-bash-3.1$ cat /etc/passwd |grep vinuesa
vinuesa:x:1058:100:Pablo Vinuesa:/home/vinuesa:/bin/bash
-bash-3.1$
```

Entre otros datos, el archivo **/etc/passwd** contiene información sobre el nombre del usuario, su home y el último dato, el programa a arrancar una vez finalizado exitosamente el login, que en este caso es el shell **bash**.

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos
- Tecleando comandos para el shell - el ciclo de comandos
 

```
-bash-3.1$ echo $PATH
/export/internal_use/marfil/bin/R-2.7.1/bin:/export/internal_use/marfil/bin/blast-2.2.16/bin:/opt/gridengine/bin/lx26-x86:/usr/X11R6/bin:/usr/kerberos/bin:/home/vinuesa/bin/rdesktop-1.5.0:/home/vinuesa/bin/Squint:/home/vinuesa/bin/treedynpack:/home/vinuesa/bin/seaview2.1
-bash-3.1$
```

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos

- Las responsabilidades del shell -

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos

- Ejecución de programas por el shell

- formato básico de un comando

```
comando [argumento1 arg2 arg3 ...]
```

```
-bash-3.1$
-bash-3.1$ cp ls.out tmp1/
-bash-3.1$
```

-bash-3.1\$ echo los espacios separan argumentos, necesito al menos 1espacio  
 los espacios separan argumentos, necesito al menos 1espacio  
 -bash-3.1\$

- Además de argumentos, los comandos pueden tener opciones, las cuales preceden a los argumentos y llevan un guión sencillo delante de una o más opciones.

```
-bash-3.1$ ls -Flsh tmp1/ # igual que ls -F -l -s -h tmp1/
total 19K
1.0K -rw-r--r-- 1 vinuesa cifn-ux 10 Jun 21 00:13 archivo1.txt
1.0K -rw-r--r-- 1 vinuesa cifn-ux 10 Jun 21 00:13 archivo2.txt
1.0K -rw-r--r-- 1 vinuesa cifn-ux 20 Jun 21 00:14 archivos2-1_concatenados.txt
15K -rw-r--r-- 1 vinuesa cifn-ux 15K Jun 21 00:44 lo_nuevo.html
1.0K -rw-r--r-- 1 vinuesa cifn-ux 130 Jun 21 2009 ls.out
-bash-3.1$
```

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

- ¿Qué máquina es ésta a la que estoy conectado?

- hostname** - read or set the hostname or the NIS domainname

```
-bash-3.1$ hostname
kayab.ccg.unam.mx
-bash-3.1$ hostname -i
132.248.34.150
-bash-3.1$
```

- uname** - Print certain system information

```
-bash-3.1$ uname
Linux
-bash-3.1$ uname -a
Linux kayab.ccg.unam.mx 2.6.18-53.1.14.el5 #1 SMP Wed Mar 5 11:36:49
EST
200886 i686 i386 GNU/Linux
-bash-3.1$
```

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

- ¿Quién soy, quien más está conectado?

- whoami** - print username

```
-bash-3.1$ whoami
vinuesa
-bash-3.1$
```

- who** - display information about the current status of the system

```
-bash-3.1$ who
root pts/1 2009-03-17 12:39 (:0.0)
vinuesa pts/2 2009-06-20 18:09 (uxmal.ccg.unam.mx)
heladia pts/3 2009-05-18 23:55 (itzamna.ccg.unam.mx)
vinuesa pts/4 2009-06-20 19:07 (uxmal.cifn.unam.mx)
rzayas pts/7 2009-06-02 10:57 (kay.ccg.unam.mx)
rzayas pts/8 2009-06-02 10:58 (kay.ccg.unam.mx)
```

```
-bash-3.1$ who am i
vinuesa pts/4 2009-06-20 19:07 (uxmal.cifn.unam.mx)
-bash-3.1$
```

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

2.- ¿Quién soy, quien más está conectado?

- `w` - print summaries of system usage, currently logged-in users and what they're doing

```
-bash-3.1$ w
19:23:18 up 102 days, 21:37, 6 users, load average: 0.07, 0.03, 0.00
USER  TTY  FROM          LOGIN@  IDLE   JCPU   PCPU   WHAT
root  pts/1  :0.0          17Mar09 78days 0.31s  0.31s  bash
vinuesa pts/2  uxm1.ccg.unam.m 18:09   1:12m  0.09s  0.09s  -bash
heladia pts/3  itzamma.ccg.unam 18May09 32days 0.71s  0.50s  -csh
vinuesa pts/4  uxm1.cifn.unam. 19:07   0.00s  0.18s  0.00s  w
rzayas pts/7  kay.ccg.unam.mx 02Jun09 15days 0.27s  0.27s  -bash
rzayas pts/8  kay.ccg.unam.mx 02Jun09 15days 0.23s  0.23s  -bash
-bash-3.1$
```

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

3.- ¿Cómo obtento información o ayuda sobre las opciones de los comandos?

- `uname -- help`

```
-bash-3.1$ uname --help
Usage: uname [OPTION]...
Print certain system information.  With no OPTION, same as -s.

-a, --all                print all information, in the following order,
                        except omit -p and -i if unknown:
-s, --kernel-name       print the kernel name
-n, --nodename          print the network node hostname
-r, --kernel-release    print the kernel release
-v, --kernel-version    print the kernel version
-m, --machine           print the machine hardware name
-p, --processor         print the processor type or "unknown"
-i, --hardware-platform print the hardware platform or "unknown"
-o, --operating-system  print the operating system
--help                 display this help and exit
--version              output version information and exit

Report bugs to <bug-coreutils@gnu.org>.
-bash-3.1$
```

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

```
[vinuesa@xibalba ~]$ printenv
MANPATH=/usr/local/qt/doc/man:/usr/local/man:/usr/local/share/man:/usr/man:/usr/share/man:
MEME_BIN=/usr/local/meme/bin
TERM=xterm
SHELL=/bin/bash
SSH_CLIENT=189.215.23.180 58428 22
PERL5LIB=/usr/local/lib/R/library/RSPer1/per1
LC_NUMERIC=en_US.ISO8859-15
MEME_DIRECTORY=/usr/local/meme
BLASTFILTER=/usr/local/wu-blast/filter
QTDIR=/usr/local/qt
SSH_TTY=/dev/pts/94
LC_ALL=es_MX
USER=vinuesa
PHRED_PARAMETER_FILE=/usr/local/phred/etc/phredpar.dat
LD_LIBRARY_PATH=/usr/local/lib/R/library/RSPer1/libs:/usr/local/lib/R/lib
RASMOLPATH=/usr/local/lib/rasmol
PLPLOT_LIB=/usr/local/emboss/lib
MAIL=/var/mail/vinuesa
PATH=/usr/local/grace/bin:/usr/jdk/latest/bin:/usr/local/blast/bin:/usr/local/mysql/bin:/usr
EULERBIN=/usr/local/euler2
LC_MESSAGES=C
LC_COLLATE=en_US.ISO8859-15
PWD=/home/vinuesa
JAVA_HOME=/usr/jdk/latest
LANG=es_MX
TZ=Mexico/General
```

### Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

3.- ¿Cómo obtento información o ayuda sobre las opciones de los comandos?

- `man uname #` (u otro comando)

```
UNAME(1)                                User Commands                                UNAME(1)

NAME
  uname - print system information

SYNOPSIS
  uname [OPTION]...

DESCRIPTION
  Print certain system information.  With no OPTION, same as -s.

-a, --all                print all information, in the following order, except omit -p and -i if
unknown:
-s, --kernel-name       print the kernel name

-n, --nodename          print the network node hostname
... salida recortada
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- explorando el nuevo ambiente ... Trabajando eficientemente con el shell
- 4.- ¿Cómo me muevo en la línea de comandos?
- Usa **ctrl-e** para ir al final de la línea
  - Usa **ctrl-a** para ir al principio de la línea
- 5.- ¿Cómo edito la línea de comandos?
- Usa la tecla **backspace** para eliminar uno a uno caracteres (del final hacia el principio)
  - Usa **ctrl-w** para eliminar una palabra completa (del final hacia el principio)
  - Usa **ctrl-u** para eliminar la línea completa (del final hacia el principio)
- 6.- ¿Cómo aborto o suspendo la ejecución de un comando?
- Usa **ctrl-c** para abortar la ejecución del último comando
- Usa **ctrl-z** para suspender la ejecución del último comando
- 7.- Repetición de la ejecución de un comando y completado de nombres de comandos/archivos
- Unix recuerda los comandos ejecutados: usa flecha arriba o abajo para moverte por el historial de comandos "history" file
  - Usa **TAB** para completar automáticamente el nombre de comandos, archivos o directorios

**Introducción al biocómputo en sistemas UNIX/Linux**

Veán también: [http://en.wikipedia.org/wiki/List\\_of\\_Unix\\_programs](http://en.wikipedia.org/wiki/List_of_Unix_programs)

- **Comandos y conceptos básicos - una selección de comandos**
- 1.- trabajando con archivos
- **ls** - lista información sobre archivos y directorio
  - **cat** - despliega contenidos de un archivo o los concatena
  - **less** - un paginador que despliega el contenido de un archivo página a página
  - **wc** - cuenta líneas, palabras y caracteres
  - **cp** - copia archivos
  - **mv** - renombra o mueve archivos
  - **rm** - elimina un archivo o directorio
  - **chmod** - cambia permisos de archivos y directorios
  - **tar** - crea un "jarro" de archivos y/o directorios
  - **zip** - comprime archivos
  - **head** - despliega la cabecera del archivo
  - **tail** - despliega la cola del archivo
  - **file** - muestra la clasificación de un archivo
- 2.- trabajando con directorios
- **pwd** - print working directory
  - **mkdir** - crea un directorio
  - **cd** - cambia de directorio
  - **rmdir** - elimina directorio (sólo si están vacíos)
  - **find** - busca archivos y directorios en base a características definidas por el usuario

**Introducción al biocómputo en sistemas UNIX/Linux**

Veán también: [http://en.wikipedia.org/wiki/List\\_of\\_Unix\\_programs](http://en.wikipedia.org/wiki/List_of_Unix_programs)

- **Comandos y conceptos básicos**
- 2.- trabajando con directorios (cont.)
- **pwd** - print working directory
  - **mkdir** - crea un directorio
  - **cd** - cambia de directorio
  - **rmdir** - elimina directorio (sólo si están vacíos)
  - **df** - despliega información de uso de disco
  - **du** - despliega información de uso de disco por archivo
- 3.- trabajando con texto
- **grep** - busca patrones en archivos
  - **cut** - selecciona caracteres o campos de archivos
  - **sort** - ordena y/o conjunta archivos
  - **uniq** - muestra líneas únicas
  - **tr** - reemplaza caracteres indicados
  - **sed** - edición no interactiva de archivos
  - **awk** - filtrado de archivos por campos
- 4.- trabajando con procesos y comandos
- **top** - despliegue dinámico de estatus de procesos
  - **ps** - despliegue de estatus de procesos
  - **kill** - mata procesos por PID
  - **nice** - cambia la prioridad de un comando
  - **which** - muestra dónde se ubica un comando en el PATH
  - **history** - muestra historial de comandos

**Introducción al biocómputo en sistemas UNIX/Linux**

- **Comandos y conceptos básicos**

## I.- trabajando con archivos

- **ls** - lista información sobre archivos y directorio

opciones importantes:

- a** display all, including hidden files .file.txt
- d** display information about directory
- l** long format
- F** classify
- h** human readable sizes
- r** reverse sorted order
- R** recursively lists subdirectories
- s** display sorted by size
- t** display sorted by creation time
- x** display files sorted by lines (default is by columns)
- 1** display files one per line

### Introducción al biocómputo en sistemas UNIX/Linux

• Comandos y conceptos básicos

Explorando la estructura de directorios con el comando ls

```
pablo@Tenerife:~$ ls -F / # exploremos el directorio raíz '/'
bin/ cdrom@ etc/ initrd.img@ lib/ lib64@ media/ opt/ root/ selinux/
sys/ usr/ vmlinuz@ xorg.conf.new boot/ dev/ home/ initrd.img.old@ lib32/
lost+found/ mnt/ proc/ sbin/ srv/ tmp/ var/ vmlinuz.old@

pablo@Tenerife:~$ ls -F /bin # veamos el contenido de /bin
bash*      bzless@  dbus-cleanup-sockets*  egrep*      kbd_mode*  ls*
bunzip2*   bzmore*  dbus-daemon*           false*     kill*       lsmmod*
busybox*   cat*     dbus-uuidgen*          fgconsole*  ksh@        mkdir*
bzip2*     chgrp*   dd*                    fgrep*     less*       mknod*
bzcmp@     chmod*   df*                    fuser*     lessecho*   mktemp*
bzdiff*    chown*   dir*                   fusermount* lessfile@   more*
bzegrep@   chvt*    dmesg*                 grep*      lesskey*    mount*
bzexe*     cp*      dnsdomainname*        gunzip*    lesspipe*   mountpoi
bzfgrep@   cpio*    domainname*            gzexe*     ln*          mt@
bzgrep*    csh@     dumpkeys*              gzip*      loadkeys*   mt-gnu*
bzip2*     dash*    echo*                  hostname*  login*      mv*
bzip2recover* date*    ed*                    ip*        lowntfs-3g* nano*
```

... y muchos más

### Introducción al biocómputo en sistemas UNIX/Linux

• Comandos y conceptos básicos

I.- trabajando con archivos - ¿dónde estoy en el árbol de directorios?

• pwd - print working directory

```
-bash-3.1$ pwd
/home/vinuesa/public_html/tlem09
-bash-3.1$ ls -lFtr
total 472
drwxr-xr-x 2 vinuesa cifn-ux 1024 Apr  3 19:21 css/
drwxr-xr-x 2 vinuesa cifn-ux 1024 Apr 13 14:12 images/
-rw----- 1 vinuesa cifn-ux 15877 May 14 01:22 index.html.save
drwxr-xr-x 2 vinuesa cifn-ux 1024 May 25 12:03 docs/
-rw-r--r-- 1 vinuesa cifn-ux 17047 Jun 10 11:36 index.html
-rw-r--r-- 1 vinuesa cifn-ux 17640 Jun 10 11:58 recursos_bioinformatica.
-rw-r--r-- 1 vinuesa cifn-ux 121862 Jun 13 14:53 posters_new.html
-rw-r--r-- 1 vinuesa cifn-ux 10061 Jun 14 00:46 profesores.html
drwxr-xr-x 2 vinuesa cifn-ux  96 Jun 15 18:08 perl_scripts/
...
-bash-3.1$
```

### Introducción al biocómputo en sistemas UNIX/Linux

• Comandos y conceptos básicos

1.- trabajando con archivos

• cat - Visualizar o concatenar el contenido de archivos

```
Usage: cat [OPTION] [FILE]...
Concatenate FILE(s), or standard input, to standard output.

-b, --number-nonblank  number nonblank output lines
-n, --number           number all output lines
-s, --squeeze-blank    never more than one single blank line
-T, --show-tabs        display TAB characters as ^I
```

- nedit es un buen editor gráfico
- vi (vim), son los editores estándar de UNIX/Linux (no gráfico), también pico y emacs
- less - es un paginador (muestra archivos por pantalla)
- more - es otro paginador (más viejo y con menor funcionalidad, por tanto: "less is more")

### Introducción al biocómputo en sistemas UNIX/Linux

• Comandos y conceptos básicos

1.- trabajando con archivos

• less - un paginador con muchas opciones (ejecuta 'man less')

```
MOVING
f ^F ^V SPACE * Forward one window (or N lines).
b ^B ESC-v    * Backward one window (or N lines).
nG           * Go TO LINE no. N

SEARCHING
/pattern     * Search forward for (N-th) matching line.
?pattern     * Search backward for (N-th) matching line.

QUIT
q

Ejemplo:
pablo@Tenerife:~$ less /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
...
```



**Introducción al biocómputo en sistemas UNIX/Linux**

• Comandos y conceptos básicos

1.- trabajando con archivos - atributos de archivos (permisos)

```
bash-3.1$ ls -lFtr
-rw-r--r-- 1 vinuesa cifn-ux 10061 Jun 14 00:46 profesores.html # - archivo
drwxr-xr-x 2 vinuesa cifn-ux 96 Jun 15 18:08 perl_scripts/ # d directorio
```

modo de acceso o permisos:  $\uparrow$  links

dueño grupo bytes Fecha y hora nombre de archivo  $\_rw-r$  modificación Nombre de directorio  $drwxr$

usuario  
grupo  
otros

Parámetros para el comando chmod (change mode)		
User	Type	Rights
u - user	+ add	r - read
g - group	- delete	w - write
o - others		x - execute
a - all		

**Introducción al biocómputo en sistemas UNIX/Linux**

• Comandos y conceptos básicos

1.- trabajando con archivos - atributos de archivos (permisos)

```
bash-3.1$ ls -lFtr
-rw-r--r-- 1 vinuesa cifn-ux 10061 Jun 14 00:46 profesores.html # - archivo
drwxr-xr-x 2 vinuesa cifn-ux 96 Jun 15 18:08 perl_scripts/ # d directorio
```

modo de acceso o permisos:  $\uparrow$  links

usuario  
grupo  
otros

Parámetros para el comando chmod (change mode)			
User	Type	Rights	
r w x	r w x	r w x	read=4
4+2+1	4+2+1	4+2+1	write=2
			execute=1

Ej. chmod 651 archivo -> lo hace u: r+w; g: r+x; o: x

**Introducción al biocómputo en sistemas UNIX/Linux**

• Comandos y conceptos básicos

1.- trabajando con archivos - atributos de archivos y cambio de permisos de acceso

```
-bash-3.1$ ls -l
drwxr-xr-x 2 vinuesa cifn-ux 1.0K Jun 20 20:13 .
drwxr-xr-x 8 vinuesa cifn-ux 1.0K Jun 20 20:13 ..
-rw-r--r-- 1 vinuesa cifn-ux 76 Jun 17 14:00 inscritos_forma_inscr.lliner
-rw-r--r-- 1 vinuesa cifn-ux 392 Jun 14 20:02 poster_lliners.txt
```

```
-bash-3.1$ chmod u+x,g-r,o-r inscritos_forma_inscr.lliner
```

```
-bash-3.1$ ls -l
-rwx----- 1 vinuesa cifn-ux 76 Jun 17 14:00 inscritos_forma_inscr.lliner
-rw-r--r-- 1 vinuesa cifn-ux 392 Jun 14 20:02 poster_lliners.txt
```

```
-bash-3.1$ ls -l
-rwx----- 1 vinuesa cifn-ux 76 Jun 17 14:00 inscritos_forma_inscr.lliner
-rw-r--r-- 1 vinuesa cifn-ux 392 Jun 14 20:02 poster_lliners.txt
```

```
-bash-3.1$ chmod 760 poster_lliners.txt
```

```
-bash-3.1$ ls -l
-rwx----- 1 vinuesa cifn-ux 76 Jun 17 14:00 inscritos_forma_inscr.lliner
-rwxrwx--- 1 vinuesa cifn-ux 392 Jun 14 20:02 poster_lliners.txt
```

```
-bash-3.1$
```

**Introducción al biocómputo en sistemas UNIX/Linux**

• Comandos y conceptos básicos

1.- trabajando con archivos - sustitución de nombres de archivos

```
-bash-3.1$ ls
inscritos_forma_inscr.lliner posters_old13Jun.html posters_old4.html
poster_lliners.txt posters_old3.html posters_old.html
```

```
-bash-3.1$ ls *txt # lista sólo los archivos terminados en .txt
poster_lliners.txt
```

```
-bash-3.1$ ls *old?.html
posters_old3.html posters_old4.html
```

```
-bash-3.1$
```

```
-bash-3.1$ ls *[0-9]*
inscritos_forma_inscr.lliner poster_lliners.txt posters_old13Jun.html
posters_old3.html posters_old4.html
```

```
-bash-3.1$ rm *[2-9]* # Elimina todos los archivos que contienen
# dígitos del 2-9 en sus nombres
```

```
-bash-3.1$ ls
posters_old.html poster_lliners.txt posters_old.html
```

```
-bash-3.1$ cp *.* ~/temp # copia los archivos que quedan a ~/temp
```

**Introducción al biocómputo en sistemas UNIX/Linux****Comandos y conceptos básicos****2. - trabajando con directorios: rutas absolutas y relativas**

- La **ruta absoluta** de un archivo o directorio es aquella que apunta hacia éste desde el directorio raíz '/', tal y como se ve en la salida del comando pwd

```
pablo@Tenerife:~/DBs$ pwd
/home/pablo/DBs
pablo@Tenerife:~/DBs$ ls
BLAST_DB  Euzeby_TAX  IMG  MLST_Ecoli_UCC  MLST_Pasteur
NCBI_GENOMES  PROFILE_DB
```

- La **ruta relativa** de un archivo o directorio es aquella que apunta hacia éste desde el directorio actual './'

```
pablo@Tenerife:~/DBs$ cd NCBI_GENOMES #cambiamos al directorio NCBI_GENOMES
pablo@Tenerife:~/DBs/NCBI_GENOMES$ cd ../.. # subimos dos niveles arriba
pablo@Tenerife:~$ pwd # el home del usuario se representa
/home/pablo/ # por ~ en UNIX/Linux
```

Es decir, para ir a /home/nombreUsuario podemos ejecutar simplemente `cd ~` ó incluso simplemente `cd`

**Introducción al biocómputo en sistemas UNIX/Linux****Comandos y conceptos básicos****2. - trabajando con directorios: rutas absolutas y relativas**

- Más ejemplos de movimientos en la estructura de directorios:

```
# estamos en /home/pablo/DBs/NCBI_GENOMES y queremos ver el contenido del
# directorio parental o superior:
pablo@Tenerife:~/DBs/NCBI_GENOMES$ ls ../
BLAST_DB  Euzeby_TAX  IMG  MLST_Ecoli_UCC  MLST_Pasteur
NCBI_GENOMES  PROFILE_DB
```

```
# estamos en /home/pablo/DBs/NCBI_GENOMES y queremos ir a ../MLST_Pasteur
# opción 1: usando la ruta relativa (al directorio actual)
pablo@Tenerife:~/DBs/NCBI_GENOMES$ cd ../MLST_Pasteur
```

```
# opción 2: usando la ruta absoluta (desde el directorio raíz)
pablo@Tenerife:~/DBs/NCBI_GENOMES$ cd ~/DBs/MLST_Pasteur
```

```
# que es lo mismo que lo de abajo:
pablo@Tenerife:~/DBs/NCBI_GENOMES$ cd /home/pablo/DBs/MLST_Pasteur
```

**Introducción al biocómputo en sistemas UNIX/Linux****Comandos y conceptos básicos****2. - trabajando con directorios: generación de directorios**

# Podemos en UNIX/Linux ejecutar varios comandos en una sola línea, separándolos con ;  
# Si la línea es muy larga, podemos introducir saltos de línea (return), escapándolo  
# con \  
# mkdir => genera directorio; no dejar espacios en blanco en el nombre, usar guión bajo!!!  
# cd => cambia al directorio  
# touch => genera archivos vacíos, para nuestra práctica  
# mv => mueve o renombra un archivo o directorio

```
pablo@Tenerife:~$ mkdir practicas_UNIX; cd practicas_UNIX; \  
touch file1.txt file2.txt file3.txt; cd ../; ls -lF practicas_UNIX;
total 0
-rw-r--r-- 1 pablo pablo 0 2011-01-15 20:13 file1.txt
-rw-r--r-- 1 pablo pablo 0 2011-01-15 20:13 file2.txt
-rw-r--r-- 1 pablo pablo 0 2011-01-15 20:13 file3.txt
pablo@Tenerife:~$ ls -d practicas_UNIX/
practicas_UNIX/
pablo@Tenerife:~$ ls practicas_UNIX/
file1.txt file2.txt file3.txt
pablo@Tenerife:~$ mkdir borrame; mv practicas_UNIX/ borrame
```

**Introducción al biocómputo en sistemas UNIX/Linux****Comandos y conceptos básicos****2. - trabajando con directorios: copiado y borrado de directorios**

```
pablo@Tenerife:~$ ls -d Pe* # muestra sólo directorios que empiecen or Pe
Perl
pablo@Tenerife:~$ ls Pe* # muestra contenido directorios que empiecen or Pe
BioPerl-1.6.0 BioPerl-1.6.0.tar.gz BioPerl_examples.tar.gz CPAN
CVS_marfil_latest examples Perl_code_move Perl_scripts_Pablo wdb.tar.Z
```

```
# ahora hacemos una copia del directorio Perl/CVS_marfil_latest y
# de su contenido al directorio borrame/
```

```
pablo@Tenerife:~$ cp -r Perl/CVS_marfil_latest/ borrame/
pablo@Tenerife:~$ ls borrame/ # vemos CVS_marfil_latest en borrame
CVS_marfil_latest file1.txt file2.txt file3.txt practicas_UNIX
```

```
# y ahora borramos el directorio borrame/ y todo su contenido
pablo@Tenerife:~$ ls -ld borrame/
drwxr-xr-x 4 pablo pablo 4096 2011-01-15 20:31 borrame/
pablo@Tenerife:~$ rm -rf borrame
pablo@Tenerife:~$ ls -ld borrame/
drwxr-xr-x 4 pablo pablo 4096 2011-01-15 20:31 borrame/
```

**Introducción al biocómputo en sistemas UNIX/Linux****Comandos y conceptos básicos****2. - trabajando con directorios: resumen**

```
# Podemos en UNIX/Linux ejecutar varios comandos en una sola línea, separándolos con ;
# Si la línea es muy larga, podemos introducir saltos de línea (return), escapándolo
# con \
# mkdir => genera directorio; no dejar espacios en blanco en el nombre, usar guión bajo!!!
# cd => cambia al directorio
# touch => genera archivos vacíos, para nuestra práctica
# mv file2 dir1; mv file3 file1 => mueve o renombra un archivo o directorio
# cp file1 dir1 => copia file1 a dir1
# cp -r dir1 dir2 => copia dir1 y su contenido (-r recursivamente) a dir2
# rm dir1/file1 => borra file1 en dir1/
# rm -rf dir1 dir2 => elimina los directorios dir1 y dir2
# find . -type d => muestra recursivamente los subdirectorios del directorio actual
# ls -d dir* => muestra sólo los nombres de los directorios que empiecen por dir
# ls dir1 => muestra el contenido del directorio dir1
```

**Introducción al biocómputo en sistemas UNIX/Linux****Comandos y conceptos básicos****2. - trabajando con directorios: copiado y borrado de directorios**

```
pablo@Tenerife:~$ ls -d Pe* # muestra sólo directorios que empiecen or Pe
Perl
pablo@Tenerife:~$ ls Pe* # muestra contenido directorios que empiecen or Pe
BioPerl-1.6.0 BioPerl-1.6.0.tar.gz BioPerl_examples.tar.gz CPAN
CVS_marfil_latest examples Perl_code_move Perl_scripts_Pablo wdb.tar.Z

# ahora hacemos una copia del directorio Perl/CVS_marfil_latest y
# de su contenido al directorio borrame/

pablo@Tenerife:~$ cp -r Perl/CVS_marfil_latest/ borrame/
pablo@Tenerife:~$ ls borrame/ # vemos CVS_marfil_latest en borrame
CVS_marfil_latest file1.txt file2.txt file3.txt practicas_UNIX

# y ahora borramos el directorio borrame/ y todo su contenido
pablo@Tenerife:~$ ls -ld borrame/
drwxr-xr-x 4 pablo pablo 4096 2011-01-15 20:31 borrame/
pablo@Tenerife:~$ rm -rf borrame
pablo@Tenerife:~$ ls -ld borrame/
drwxr-xr-x 4 pablo pablo 4096 2011-01-15 20:31 borrame/
```

**Introducción al biocómputo en sistemas UNIX/Linux****Comandos y conceptos básicos - I/O****3.- Standard Input / Standard Output**

```
# salida del comando who a STDOUT (pantalla por lo general)
-bash-3.1$ who
root pts/1 2009-03-17 12:39 (:0.0)
heladia pts/3 2009-05-18 23:55 (itzamna.ccg.unam.mx)
vinuesa pts/4 2009-06-20 19:36 (uxmal.ccg.unam.mx)
rzayas pts/7 2009-06-02 10:57 (kay.ccg.unam.mx)
rzayas pts/8 2009-06-02 10:58 (kay.ccg.unam.mx)

# redireccionamos la salida de who a un archivo con >
-bash-3.1$ who > users.out

# podemos ver el contenido de users.out con cat o less
-bash-3.1$ cat users.out
root pts/1 2009-03-17 12:39 (:0.0)
heladia pts/3 2009-05-18 23:55 (itzamna.ccg.unam.mx)
vinuesa pts/4 2009-06-20 19:36 (uxmal.ccg.unam.mx)
rzayas pts/7 2009-06-02 10:57 (kay.ccg.unam.mx)
rzayas pts/8 2009-06-02 10:58 (kay.ccg.unam.mx)
-bash-3.1$
```

**Introducción al biocómputo en sistemas UNIX/Linux****Comandos y conceptos básicos - I/O****3.- Standard Input / Standard Output**

```
# uso del comando cat para conCATenar texto o archivos al final de otro usando >>
-bash-3.1$ cat >> users.out
estas son lineas adicionales al final del archivo gracias a '>>'
y una segunda linea
y una tercera

# veamos el contenido de users.out con less
bash-3.1$ less < users.out # (equivalente a less users.out)
root pts/1 2009-03-17 12:39 (:0.0)
heladia pts/3 2009-05-18 23:55 (itzamna.ccg.unam.mx)
vinuesa pts/4 2009-06-20 19:36 (uxmal.ccg.unam.mx)
rzayas pts/7 2009-06-02 10:57 (kay.ccg.unam.mx)
rzayas pts/8 2009-06-02 10:58 (kay.ccg.unam.mx)
estas son lineas adicionales al final del archivo gracias a '>>'
y una segunda linea
y una tercera
(END)

# de esta manera añadimos el contenido de file1 al final de users.out
-bash-3.1$ cat file1 >> users.out
```

**Introducción al biocómputo en sistemas UNIX/Linux****• Comandos y conceptos básicos - I/O****3.- Standard Input / Standard Output - más ejemplos de I/O con cat**

```
# el comando echo imprime a STOUT su argumento(s); redirigimos salida a archivo1.txt
# simplemente para tener un archivo de texto con contenido
-bash-3.1$ echo 'línea uno' > archivo1.txt
-bash-3.1$ less archivo1.txt
línea uno

# lo mismo lo podemos hacer con el comando cat, como ya hemos visto anteriormente
-bash-3.1$ cat > archivo2.txt
línea dos
^D # usen CTRL-D para interrumpir la escritura a archivo2.txt con cat
-bash-3.1$ cat archivo2.txt archivo1.txt # concatena ambos archivos
línea dos
línea uno

# redirigimos con '>' la salida de cat a un archivo
-bash-3.1$ cat archivo2.txt archivo1.txt > archivos2-1_concatenados.txt

-bash-3.1$ cat archivos2-1_concatenados.txt
línea dos
línea uno
```

**Introducción al biocómputo en sistemas UNIX/Linux****• Comandos y conceptos básicos - Pipes (tuberías)****4.- pipes '|' - conecta la salida (stdout) de un comando directamente con el stdin de otro comando, filtrando la salida del primero por el segundo programa**  
Típicos programas de filtrado son grep, cut, sed, awk, head, tail, wc, perl ...

```
-bash-3.1$ who
root pts/1 2009-03-17 12:39 (:0.0)
heladia pts/3 2009-05-18 23:55 (itzamna.ccg.unam.mx)
vinuesa pts/4 2009-06-20 19:36 (uxmal.ccg.unam.mx)
rzayas pts/7 2009-06-02 10:57 (kay.ccg.unam.mx)
rzayas pts/8 2009-06-02 10:58 (kay.ccg.unam.mx)
-bash-3.1$ who | wc -l # cuenta las líneas de la salida de who con 'word count -l'
5
bash-3.1$ who | head -2 # dame sólo las 2 las. Líneas de la salida de who con 'head -2'
root pts/1 2009-03-17 12:39 (:0.0)
heladia pts/3 2009-05-18 23:55 (itzamna.ccg.unam.mx)
-bash-3.1$ who | tail -3 # dame sólo las 3 líneas finales de who con 'tail -3'
vinuesa pts/4 2009-06-20 19:36 (uxmal.ccg.unam.mx)
rzayas pts/7 2009-06-02 10:57 (kay.ccg.unam.mx)
rzayas pts/8 2009-06-02 10:58 (kay.ccg.unam.mx)
-bash-3.1$
```

**Introducción al biocómputo en sistemas UNIX/Linux****• Comandos y conceptos básicos - herramientas de filtrado usadas en pipes****7.- Veremos las siguientes herramientas: grep, cut, sort y uniq (exploren uds. tr y sed)****• grep**

```
-bash-3.1$ grep --help # selección de opciones (hay muchas más)
Usage: grep [OPTION]... PATTERN [FILE] ...
Search for PATTERN in each FILE or standard input.
Example: grep -i 'hello world' menu.h main.c

Regexp selection and interpretation:
-P, --perl-regexp PATTERN is a Perl regular expression
-e, --regexp=PATTERN use PATTERN as a regular expression
-f, --file=FILE obtain PATTERN from FILE
-i, --ignore-case ignore case distinctions
-w, --word-regexp force PATTERN to match only whole words
-x, --line-regexp force PATTERN to match only whole lines

Miscellaneous:
-v, --invert-match select non-matching lines

Output control:
-n, --line-number print line number with output lines
--line-buffered flush output on every line
-L, --files-without-match only print FILE names containing no match
-l, --files-with-matches only print FILE names containing matches
```

**Introducción al biocómputo en sistemas UNIX/Linux****• Comandos y conceptos básicos - herramientas de filtrado usadas en pipes****7.- Veremos las siguientes herramientas: grep, cut, sort y uniq (exploren uds. tr y sed)****• cut**

```
-bash-3.1$ cut --help # selección de opciones (hay muchas más)

Usage: cut [OPTION]... [FILE]...
Print selected parts of lines from each FILE to standard output.

Mandatory arguments to long options are mandatory for short options too.
-c, --characters=LIST select only these characters
-d, --delimiter=DELIM use DELIM instead of TAB for field delimiter
-f, --fields=LIST select only these fields; also print any line
that contains no delimiter character, unless
the -s option is specified
With no FILE, or when FILE is -, read standard input.
```

**Introducción al biocómputo en sistemas UNIX/Linux**

## • Comandos y conceptos básicos - herramientas de filtrado usadas en pipes

7.- Veremos las siguientes herramientas: `grep`, `cut`, `sort` y `uniq` (exploren uds. `tr` y `sed`)• **sort**

```
-bash-3.1$ sort --help # selección de opciones (hay algunas más)
Usage: sort [OPTION]... [FILE]...
Write sorted concatenation of all FILE(s) to standard output.
```

Mandatory arguments to long options are mandatory for short options too.  
Ordering options:

```
-b, --ignore-leading-blanks  ignore leading blanks
-d, --dictionary-order       consider only blanks and alphanumeric characters
-f, --ignore-case            fold lower case to upper case characters
-g, --general-numeric-sort   compare according to general numerical value
-I, --ignore-nonprinting     consider only printable characters
-M, --month-sort            compare (unknown) < `JAN' < ... < `DEC'
-n, --numeric-sort          compare according to string numerical value
-r, --reverse                reverse the result of comparisons
```

Other options:

```
-m, --merge                merge already sorted files; do not sort
-o, --output=FILE          write result to FILE instead of standard output
-t, --field-separator=SEP  use SEP instead of non-blank to blank transition

-u, --unique                with -c, check for strict ordering;
                           without -c, output only the first of an equal run
```

**Introducción al biocómputo en sistemas UNIX/Linux**

## • Comandos y conceptos básicos - herramientas de filtrado usadas en pipes

• **uniq**

```
-bash-3.1$ uniq --help
Usage: uniq [OPTION]... [INPUT [OUTPUT]]
Discard all but one of successive identical lines from INPUT (or
standard input), writing to OUTPUT (or standard output).
```

Mandatory arguments to long options are mandatory for short options too.

```
-c, --count                prefix lines by the number of occurrences
-d, --repeated             only print duplicate lines
-D, --all-repeated[=delimit-method] print all duplicate lines
                           delimit-method={none(default),prepend,separate}
                           Delimiting is done with blank lines.
-f, --skip-fields=N        avoid comparing the first N fields
-i, --ignore-case          ignore differences in case when comparing
-s, --skip-chars=N        avoid comparing the first N characters
-u, --unique               only print unique lines
-w, --check-chars=N       compare no more than N characters in lines
--help                    display this help and exit
--version                 output version information and exit
```

A field is a run of whitespace, then non-whitespace characters.  
Fields are skipped before chars.

**Introducción al biocómputo en sistemas UNIX/Linux**

## • Comandos y conceptos básicos - herramientas de filtrado usadas en pipes

• Veamos unas tuberías en acción - analizando el archivo `/etc/passwd`

```
-bash-3.1$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
# ... muchas líneas cortadas
lsayaved:x:1635:104:Lizbeth Sayavedra Camacho:/home/lsayaved:/bin/bash
fsanchez:x:15031:104:Federico A. Sanchez Quinto:/home/fsanchez:/bin/bash
mmatus:x:1546:104:Mariana Guadalupe Matus Garcia:/home/mmatus:/bin/bash
jmorales:x:1622:104:Jose Alejandro Morales Tapia:/home/jmorales:/bin/bash
trosas:x:15029:104:Tania Rosas Perez:/home/trosas:/bin/bash
rsat:x:1003:100:Administrador RSAT:/home/rsat:/bin/tcsh
lservin:x:1559:104:Luis Eduardo Servin Garciduenas:/home/lservin:/bin/bash
jvaldes:x:20170:100:Jorge Valdes:/home/jvaldes:/bin/bash
```

- ¿Cuántas líneas contiene el archivo?

```
-bash-3.1$ cat /etc/passwd |wc -l
77
-bash-3.1$ wc -l /etc/passwd
77 /etc/passwd
```

**Introducción al biocómputo en sistemas UNIX/Linux**

## • Comandos y conceptos básicos - herramientas de filtrado usadas en pipes

• Veamos unas tuberías en acción - analizando el archivo `/etc/passwd`

- ¿Cuántos usuarios hay en el sistema (nota, todos tiene asignado un home)

```
-bash-3.1$ grep home /etc/passwd
rzayas:x:8028:3001:Romualdo Zayas Lagunas:/home/rzayas:/bin/bash
amedina:x:15020:3001:Alejandra E. Medina Rivera:/home/amedina:/bin/bash
jvanheld:x:1004:101:Jacques van Helden:/home/jvanheld:/bin/tcsh
espinosa:x:20063:3001:Vladimir Espinosa:/home/espinosa:/bin/bash
vinuesa:x:1058:100:Pablo Vinuesa:/home/vinuesa:/bin/bash
llopez:x:20057:100:Lucia N. Lopez:/home/llopez:/bin/bash
jesquive:x:1537:104:Julian Antonio Esquivel Marquez:/home/jesquive:/bin/bash
aavila:x:15001:104:Agustin B. Avila Casanueva:/home/aavila:/bin/bash
# ... muchas líneas cortadas
-bash-3.1$ grep -c home /etc/passwd
35
-bash-3.1$
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- Comandos y conceptos básicos - herramientas de filtrado usadas en pipes
- Veamos unas tuberías en acción - analizando el archivo /etc/passwd

- Imprime una lista ordenada del userID y del nombre completo del usuario

```
-bash-3.1$ grep home /etc/passwd |cut -d: -f1,5 |sort
aavila:Agustin B. Avila Casanueva
ajhernan:Alfredo J. Hernandez Alvarez
amedina:Alejandra E. Medina Rivera
amedrano:Arturo Medrano
aroth:Alexandra Jazmin Roth Schulze
# ... 35 entradas en total.
```

- Para estos 35 usuarios, obtén una estadística de preferencia de shell

```
-bash-3.1$ grep home /etc/passwd |cut -d: -f7 |sort |uniq -c
   30 /bin/bash
    5 /bin/tcsh
-bash-3.1$
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- Comandos y conceptos básicos - Introducción a la programación en bash
- Uso de variables e impresión de su contenido desde línea de comandos
- bash-3.1\$ STR='Hello World'; echo \$STR
- Uso de condicionales y su ejecución desde un "script". Hacerlo ejecutable con chmod +x script

```
#!/bin/bash
# program: simple_conditionals.sh

# 1)check that two arguments are passed to the script from the command line
if [ $# != 2 ]; then
    echo "# $0 needs two string arguments to compare"
    echo "# usage: $0 string1 string2"
    exit 1
fi

# 2) assign positional parameters to named variables
string1="$1"
string2="$2"

# 3) make the string comparisons within an if-else-fi structure
if [ "$string1" = "$string2" ]; then
    echo "$string1 = $string2, therefore expression evaluated as true"
else
    echo "$string1 != $string2, therefore expression evaluated as false"
fi
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- Comandos y conceptos básicos - Introducción a la programación en bash
- Uso de bucles y condicionales desde un script

```
#!/bin/bash
# program: find_directories.sh

# 1) inicializamos variables; var=$(comando) guarda salida de comando UNIX en var
workdir=$(pwd)
counter=0

# 2) recorremos cada archivo/dir en pwd; si es un dir, imprimimos e incrementamos contador
for i in $( ls ); do
    if [ -d $i ]; then
        echo found directory: $i
        let counter=counter+1
    fi
done

# 3) evaluamos el contenido de la variable contador e imprimimos resumen correspondiente
if [ $counter = 0 ]; then
    echo "There are no directories in $workdir"
    exit 0
elif [ $counter > 0 ]; then
    echo "There are There are $counter directories in $workdir"
fi
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- Comandos y conceptos básicos - Introducción a la programación en bash
- Uso de bucles y condicionales desde la línea de comandos

```
for i in $(ls); do if [ -f $i ]; then echo file $i; elif [ -d $i ]; then echo dir $i; fi; done
```

```
# SALIDA
file find_directories.sh
file lista_login_accounts.tab
file lista_login_accounts.txt
file parse_seleccionados1_html.1liners
file samble_bashrc.txt
file seleccionados1.html
file simple_conditionals.sh
dir tmp
```

- El uso de bucles y condicionales desde la línea de comandos puede ser muy útil. Este ejemplo alinea todos los archivos fasta con terminación fna presentes en el directorio actual, usando muscle

```
for file in *.fna; do muscle < $file > ${file%.fna}_muscle_alignment.fna; done
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- Comandos y conceptos básicos - miscelánea de comandos muy importantes
- generar un tarro (tar file o "carpeta") , añadirle archivos, comprimir el tarro y enviarlo a una máquina remota mediante scp:

```
# vamos a generar un tarro comprimido con gzip, que contenga todos los *.sh scripts
# presentes en el directorio actual
```

```
pablo@Tenerife:~/WinXP/Cursos/TLEM11$ ls *sh
align_seqs_clustal_or_muscle.sh find_directories.sh simple_conditionals.sh
pablo@Tenerife:~/WinXP/Cursos/TLEM11$ tar -cvzf sample_bash_scripts.tgz *sh
align_seqs_clustal_or_muscle.sh
find_directories.sh
simple_conditionals.sh
```

```
# ahora vamos a copiar el tarro comprimido mediante scp de mi una máquina a otra
```

```
pablo@Tenerife:~/WinXP/Cursos/TLEM11$ ls *tgz
sample_bash_scripts.tgz
pablo@Tenerife:~/WinXP/Cursos/TLEM11$ scp sample_bash_scripts.tgz \
vinuesa@132.248.34.3:/home/vinuesa/public_html/tlem
sample_bash_scripts.tgz      100% 1485    1.5KB/s   00:00
pablo@Tenerife:~/WinXP/Cursos/TLEM11$
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- Comandos y conceptos básicos - miscelánea de comandos muy importantes
- bajar archivos de la web desde la terminal:

```
# si no lo han hecho aún, generen los siguientes directorios en su $HOME
```

```
mkdir practicas_unix; mkdir bin; mkdir seq_data; cd bin;
```

```
[vinuesa@xibalba ~]$ cd bin/
```

```
[vinuesa@xibalba bin]$ wget -c \
```

```
http://www.ccg.unam.mx/~vinuesa/tlem/shell\_scripts/sample\_bash\_scripts.tgz
```

```
--19:56:13--
```

```
http://www.ccg.unam.mx/~vinuesa/tlem/shell_scripts/sample_bash_scripts.tgz
```

```
=> `sample_bash_scripts.tgz'
```

```
Resolving www.ccg.unam.mx... 132.248.34.17
```

```
Connecting to www.ccg.unam.mx|132.248.34.17|:80... connected.
```

```
HTTP request sent, awaiting response... 200 OK
```

```
Length: 1,485 (1.5K) [application/x-gzip]
```

```
100%[=====] 1,485    ---K/s
```

```
19:56:13 (59.26 MB/s) - `sample_bash_scripts.tgz' saved [1485/1485]
```

```
# ahora podemos desempacar y descomprimir los archivos contenidos en el tarro comprimido
```

```
[vinuesa@xibalba bin]$ tar -xvzf sample_bash_scripts.tgz
```

```
align_seqs_clustal_or_muscle.sh
```

```
find_directories.sh
```

```
simple_conditionals.sh
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- Comandos y conceptos básicos - miscelánea de comandos muy importantes
- bajar archivos de la web desde la terminal:

```
# y ahora vamos a bajar un archivo de configuración ".bashrc" del ambiente para
```

```
# que puedan trabajar más agusto en su casa $HOME
```

```
# vayan a su directorio home
```

```
cd ~;
```

```
[vinuesa@xibalba ~]$ wget -c \
```

```
http://www.ccg.unam.mx/~vinuesa/tlem/docs/sample\_bashrc.txt
```

```
Resolving www.ccg.unam.mx... 132.248.34.17
```

```
Connecting to www.ccg.unam.mx|132.248.34.17|:80... connected.
```

```
HTTP request sent, awaiting response... 200 OK
```

```
Length: 1,812 (1.8K) [text/plain]
```

```
100%[=====] 1,812    ---K/s
```

```
20:51:52 (63.30 MB/s) - `sample_bashrc.txt' saved [1812/1812]
```

```
# ahora lo renombramos a .bashrc
```

```
[vinuesa@xibalba ~]$ mv sample_bashrc.txt .bashrc
```

```
# y activamos el ambiente, ejecutando el script de configuración
```

```
[vinuesa@xibalba ~]$ source .bashrc .profile
```

**Introducción al biocómputo en sistemas UNIX/Linux**

- Referencias sobre Shell y Bash libremente disponibles en la web

```
# una lista de comandos y ejemplos de uso los encuentras aquí:
```

```
- http://en.wikipedia.org/wiki/List\_of\_Unix\_programs
```

```
# Estos son unos tutoriales que si los estudias te harán un experto programador
```

```
# de Bash. Comienza por el primero de ellos, que es muy corto. El 2°. Se basa
```

```
# en tcsh. Los últimos 2 son tutoriales avanzados sobre Bash scripting
```

```
- http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html
```

```
- http://www.ee.surrey.ac.uk/Teaching/Unix/
```

```
- http://tldp.org/LDP/Bash-Beginners-Guide/html/index.html
```

```
- http://www.museum.state.il.us/ismdepts/library/linuxguides/abs-guide/index.html
```

```
# Si buscas libros de referencia, mira aquí
```

```
- http://en.wikibooks.org/wiki/Guide\_to\_Unix
```

```
- http://www.ee.surrey.ac.uk/Teaching/Unix/books-uk.html
```