

# Desarrollo de Software en Astrofísica

Sala 763 Pabellón Forma, Profesor Sebastián Pérez  
28 de Marzo 2023

# Introducción a UNIX

(para astrónomos)

- Qué es un sistema operativo (OS)?

# Introducción a UNIX

(para astrónomos)

- Qué es un sistema operativo (OS)?
- Un set de programas que actua como interface entre usuarios y el hardware de un computador
- Provee un “ambiente” en el cual un usuario puede ejecutar programas
- Administra la memoria, los procesadores, los “devices”, la información, etc.
- Ejemplos: **Unix**, MS DOS, Microsoft Windows, **macOS**, **Linux**.

# Introducción a UNIX

(para astrónomos)

- Qué es UNIX? UNIX es una familia de sistemas operativos que ha sido desarrollada desde la década de 1960. Es ampliamente utilizado en servidores, estaciones de trabajo y supercomputadoras debido a su estabilidad, seguridad y flexibilidad.
- A diferencia de Windows, que tiene una interfaz gráfica de usuario (GUI), los sistemas UNIX suelen utilizar una interfaz de línea de comandos (CLI) para interactuar con el sistema operativo.
- La interfaz de línea de comandos puede parecer abrumadora al principio, pero permite un control más preciso sobre el sistema y es particularmente útil para tareas de administración del sistema.
- Los usuarios interactúan con el sistema ingresando comandos de texto en un emulador de terminal, en lugar de usar un mouse y elementos gráficos. Una de las fortalezas principales de UNIX es su filosofía de "hacer una cosa y hacerla bien".



Ken Thompson y Dennis Ritchie

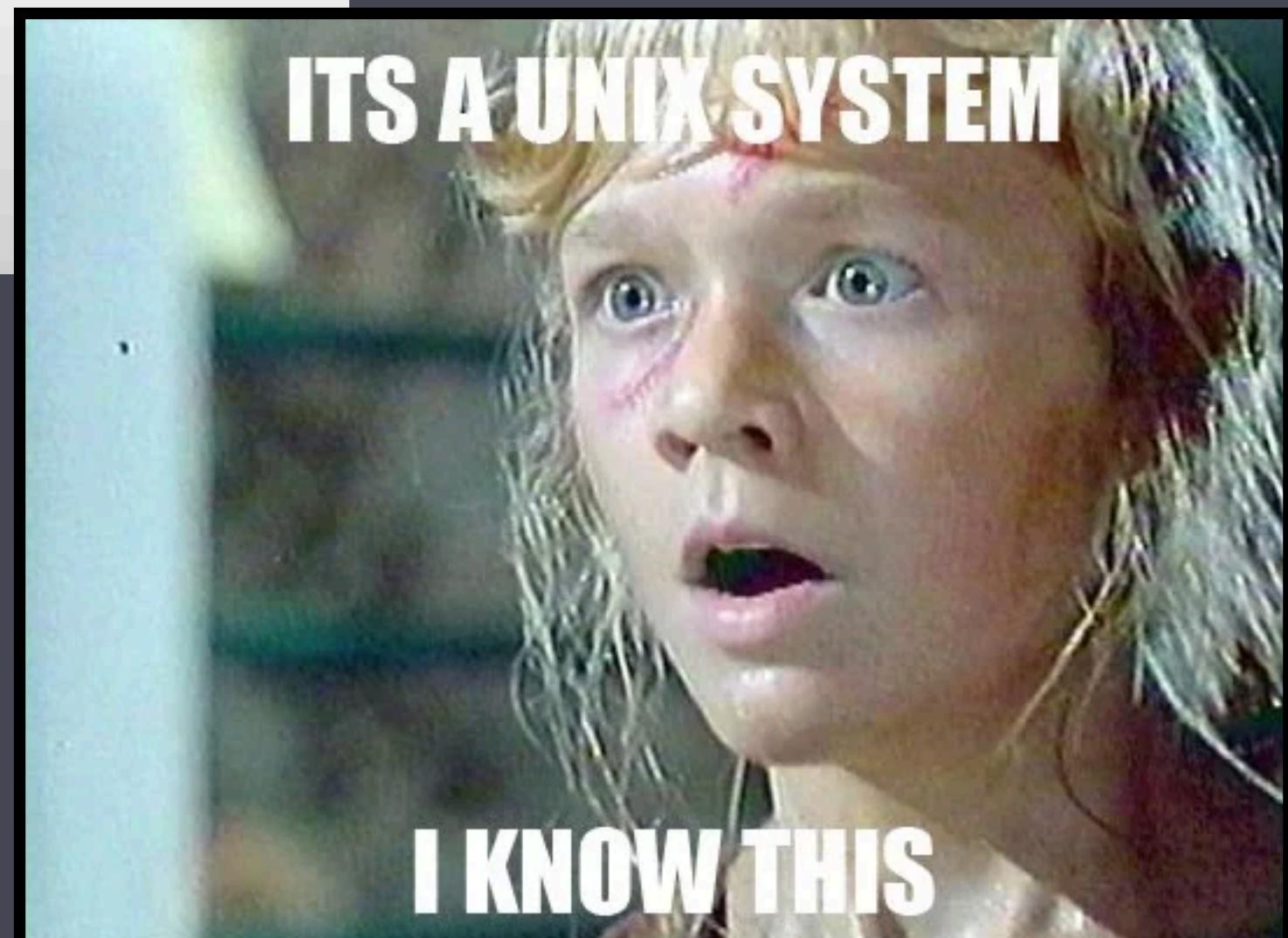
# Features of Unix

Unix Operating System supports the following basic features and capabilities :

- **Multi-user** : More than one user can use the machine simultaneously supported via terminals
- **Multi-tasking** : Multiple programs can be run at a time
- **Multi-process** : Each user can execute several processes simultaneously
- **Hierarchical Structure** : Unix directories are present like a tree structure to support the organization and maintenance of files
- **Open System** : Some of the Unix OS are open-source. Users can modify the Unix source code
- **Portability** : It is the ability of the software that operates from one machine to another machine having different configuration

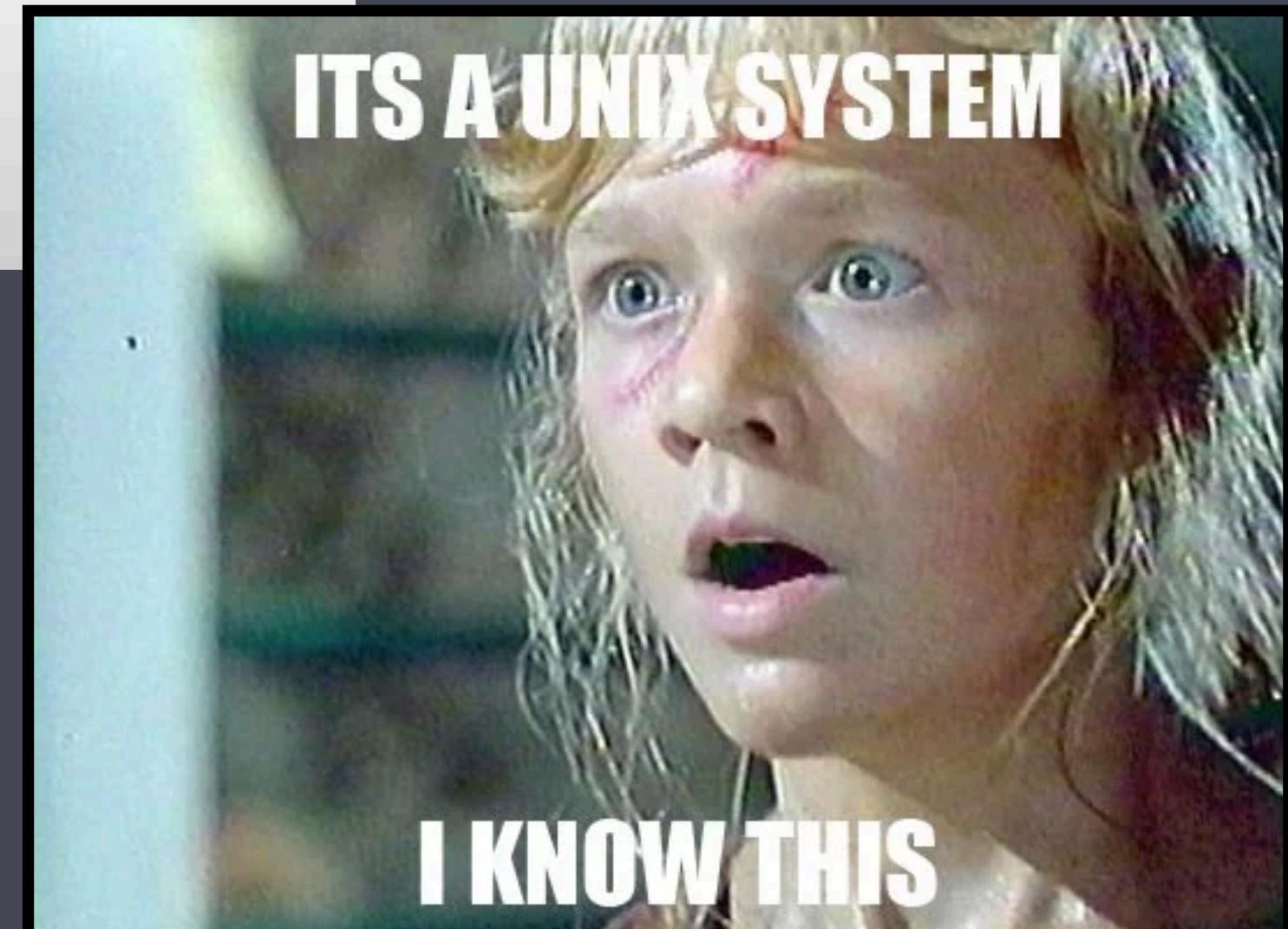
Unix allows users to transfer data from one system to another

- **Programming Facility** : Unix Shell can be used as a Programming/Scripting Language



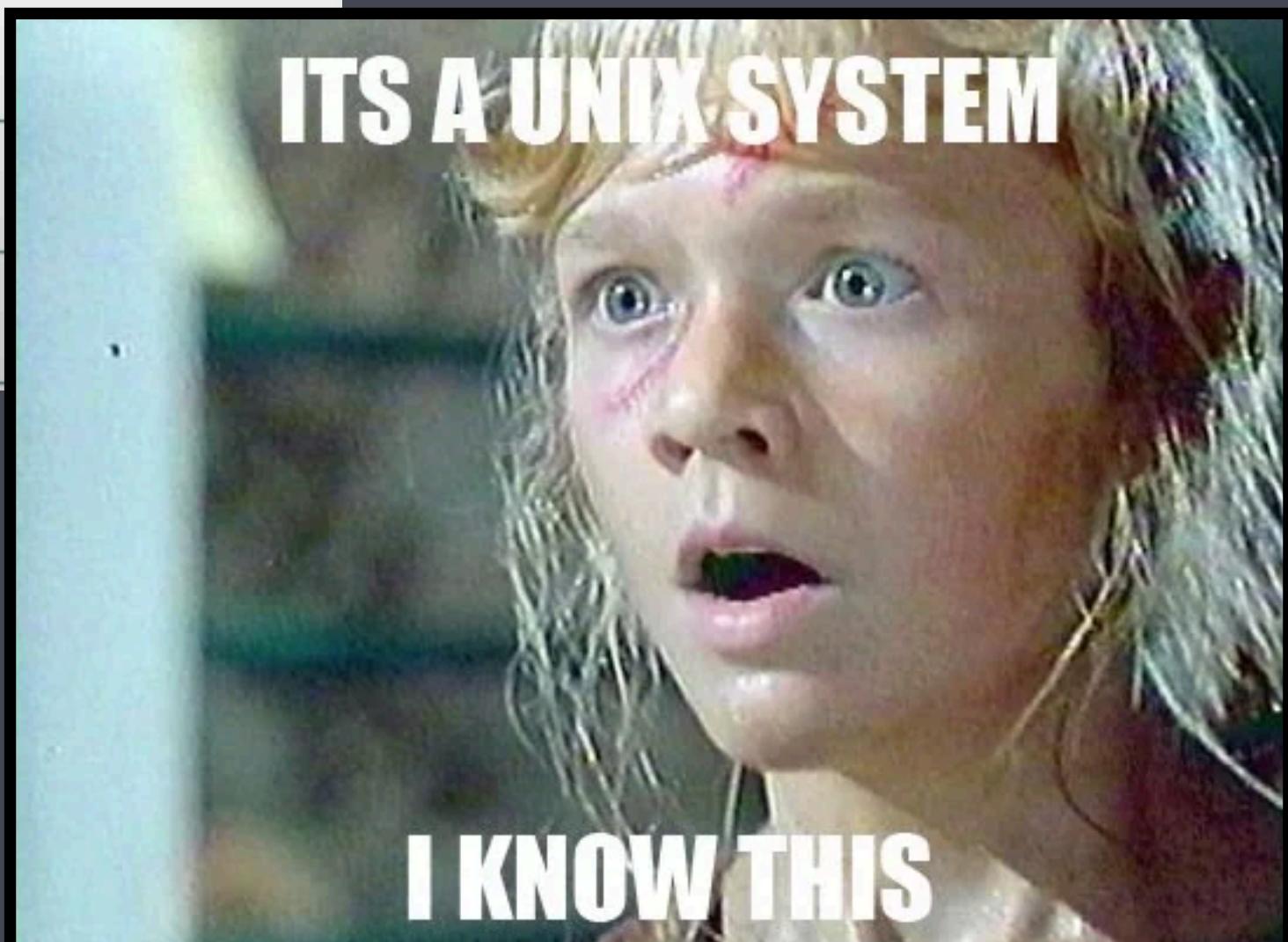
# Features of Unix (Cont.)

- **Communication Facility** : Unix allows communication between different users by providing some information
- **Security** : Unix has system level security controlled by system administrator and file level security controlled by owner of the file
- **Tools and Utilities** : Supports many of the tools, libraries and utilities to aid software development
- **Piping** : In Piping concept, the output of the first command becomes the input of the next command/process
- **Help Facility** : In Unix, 'man' command is used to view help content on any command
- **Modularity** : Unix consists of multiple number of independent modules or programs which perform different elementary tasks



# Differences Between Unix and MS-Windows

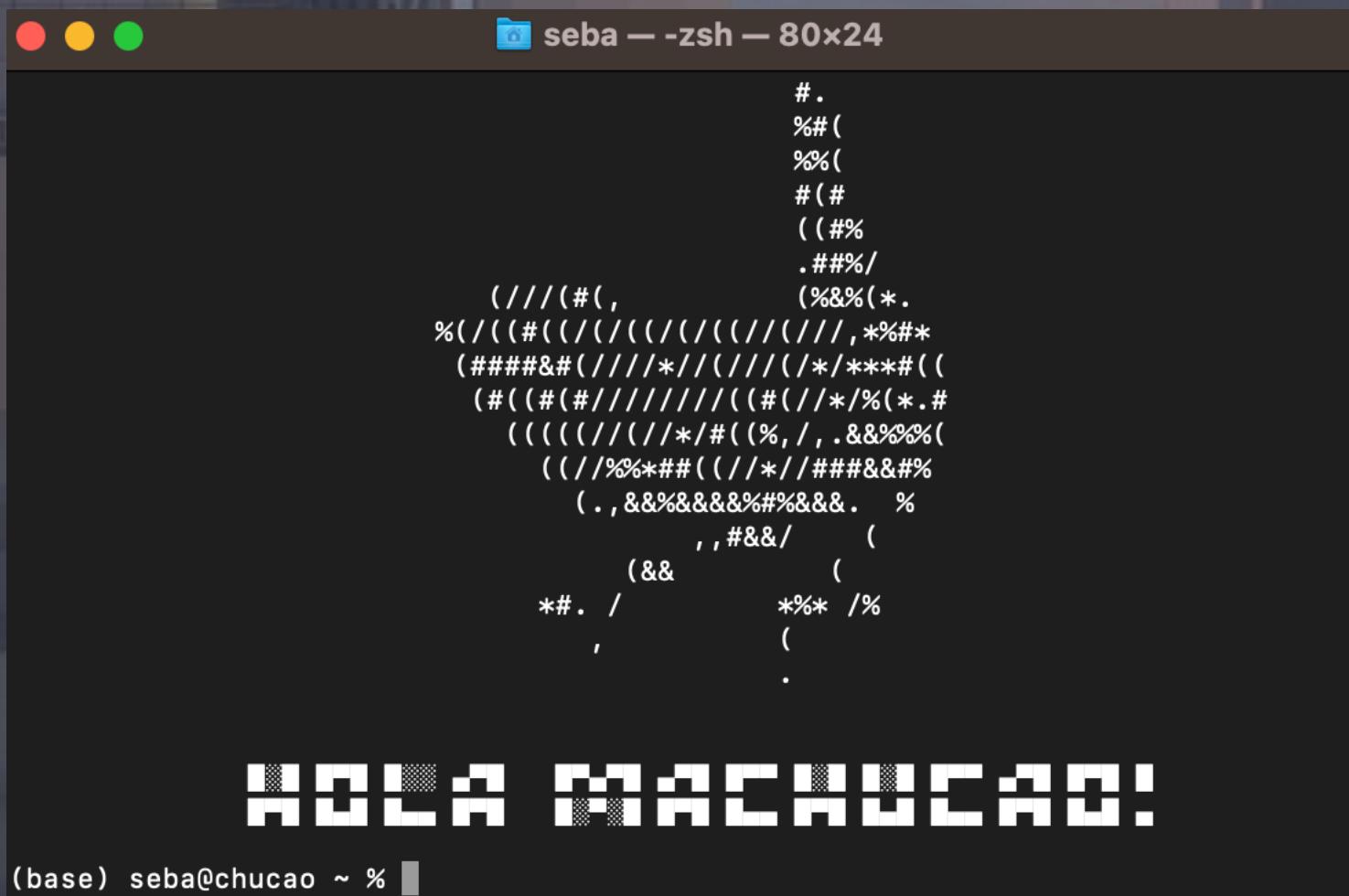
Unix	MS-Windows
Unix file system is a hierarchical model	Windows file system is a flat model
Unix is a CUI (Command User Interface) OS	Windows is a GUI (Graphical User Interface) OS
Unix is a multi-user and multi-tasking OS	Windows is a single-user and multi-tasking OS
Unix is a free-source OS	Windows is a licensed OS
Unix has dumb terminals (without hard disk)	Windows do not support dumb terminals
Unix is case sensitive	Windows is not case sensitive
Unix is not user friendly	Windows is user friendly
Unix supports programming facility	Windows do not support programming facility
Unix have multiple vendors	Windows have only one vendor; i.e. Microsoft



# Introducción a UNIX

(para astrónomos)

- Hoy la mayoría de las/os astrónomos usan un OS que sea “unix-like”, como linux o macOS.
- Usuarios le dicen “qué hacer” al computador a través de una interfaz: el terminal (shell/command line/consola/prompt)



```
seba -- zsh -- 80x24
.
%#
%%
#(
(#
(#
.#
.#
(%&(*.
%(/((#(),
%(/((#(((//((//,*%#*
(%##&#(/////*//((///*/**#(((
(#((#(////////((#(///*%(*.#
((((///*#((%,/.&&%#%
((//%#*(///*//###&#%
(.,&%&&%#&#&%
(,
(#
(#
*&.
/
*%* /%
,
.
HOLA MACHUCAO!
(base) seba@chucao ~ %
```



**ls**: Lista los archivos y directorios en el directorio actual.

Ejemplo: **ls**

**cd**: Cambia al directorio actual.

Ejemplo: **cd /home/user/documents**

**mkdir**: Crea un nuevo directorio.

Ejemplo: **mkdir mi\_carpeta**

**rm**: Elimina un archivo o directorio.

Ejemplo: **rm mi\_archivo.txt**

**cp**: Copia un archivo o directorio.

Ejemplo: **cp archivo.txt /home/user/documents**

**mv**: Mueve un archivo o directorio.

Ejemplo: **mv archivo.txt /home/user/documents**

**grep**: Busca un patrón en un archivo.

Ejemplo: **grep "patrón" archivo.txt**

**cat**: Concatena y muestra el contenido de un archivo.

Ejemplo: **cat archivo.txt**

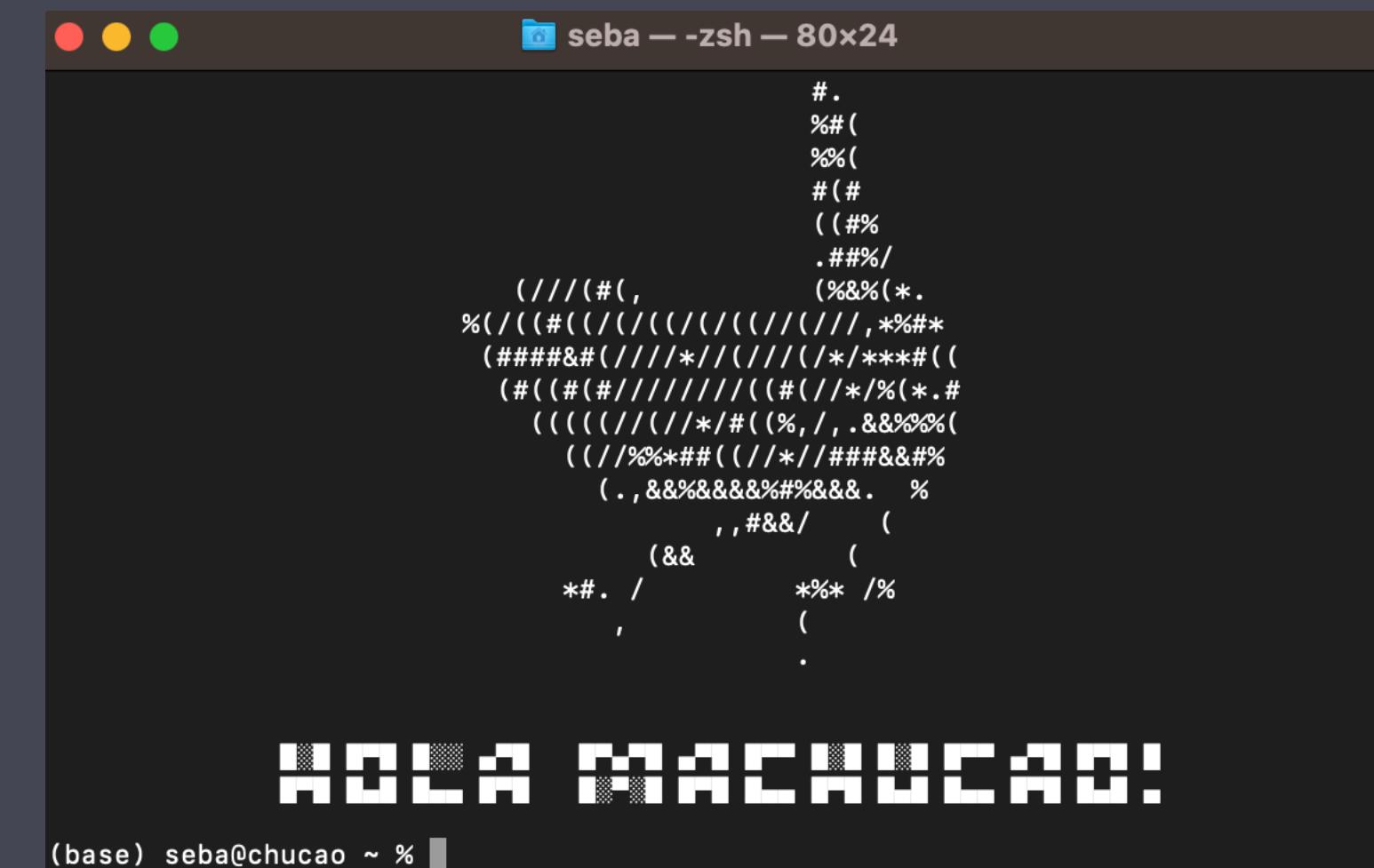
**echo**: Muestra un mensaje en la pantalla.

Ejemplo: **echo "¡Hola, mundo!"**

**TURN OFF THE GUI LUKE**



**USE THE COMMAND LINE**



1. `ls | grep "mi_archivo"`: Lista todos los archivos en el directorio actual y filtra solo los que contienen la cadena "mi\_archivo".

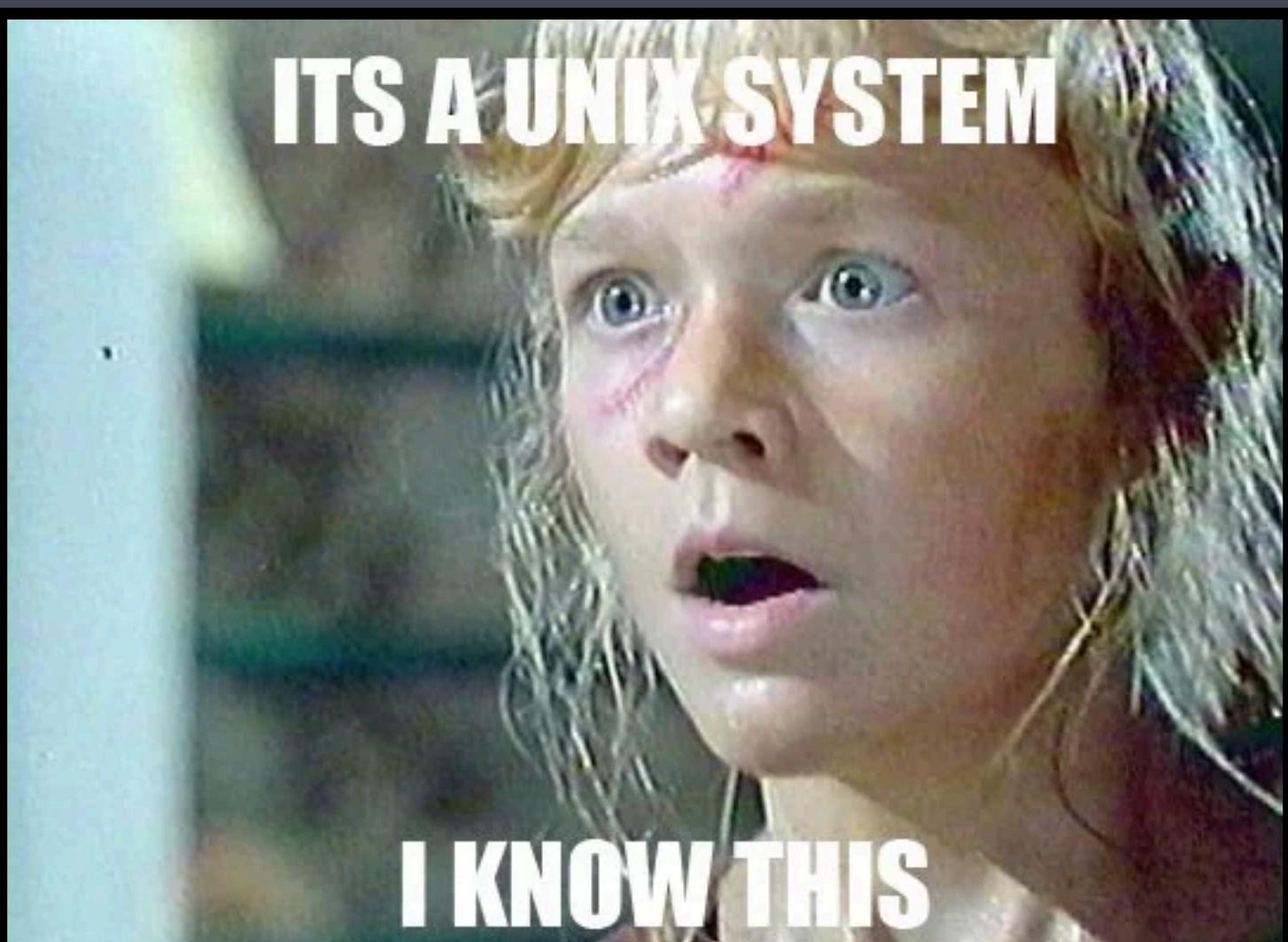
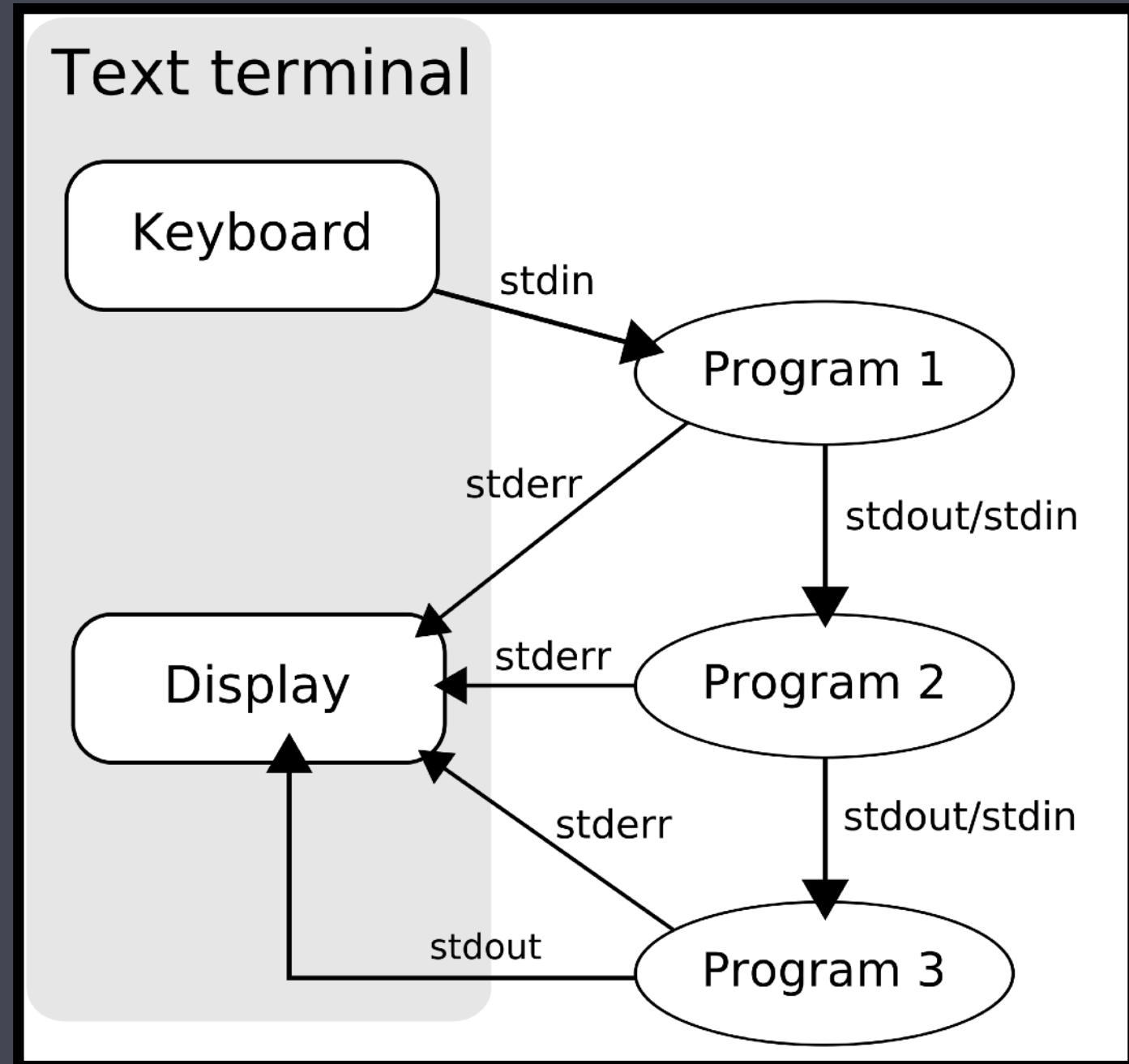
2. `cat archivo.txt | grep "patrón"`: Muestra el contenido del archivo archivo.txt y filtra solo las líneas que contienen la cadena "patrón".

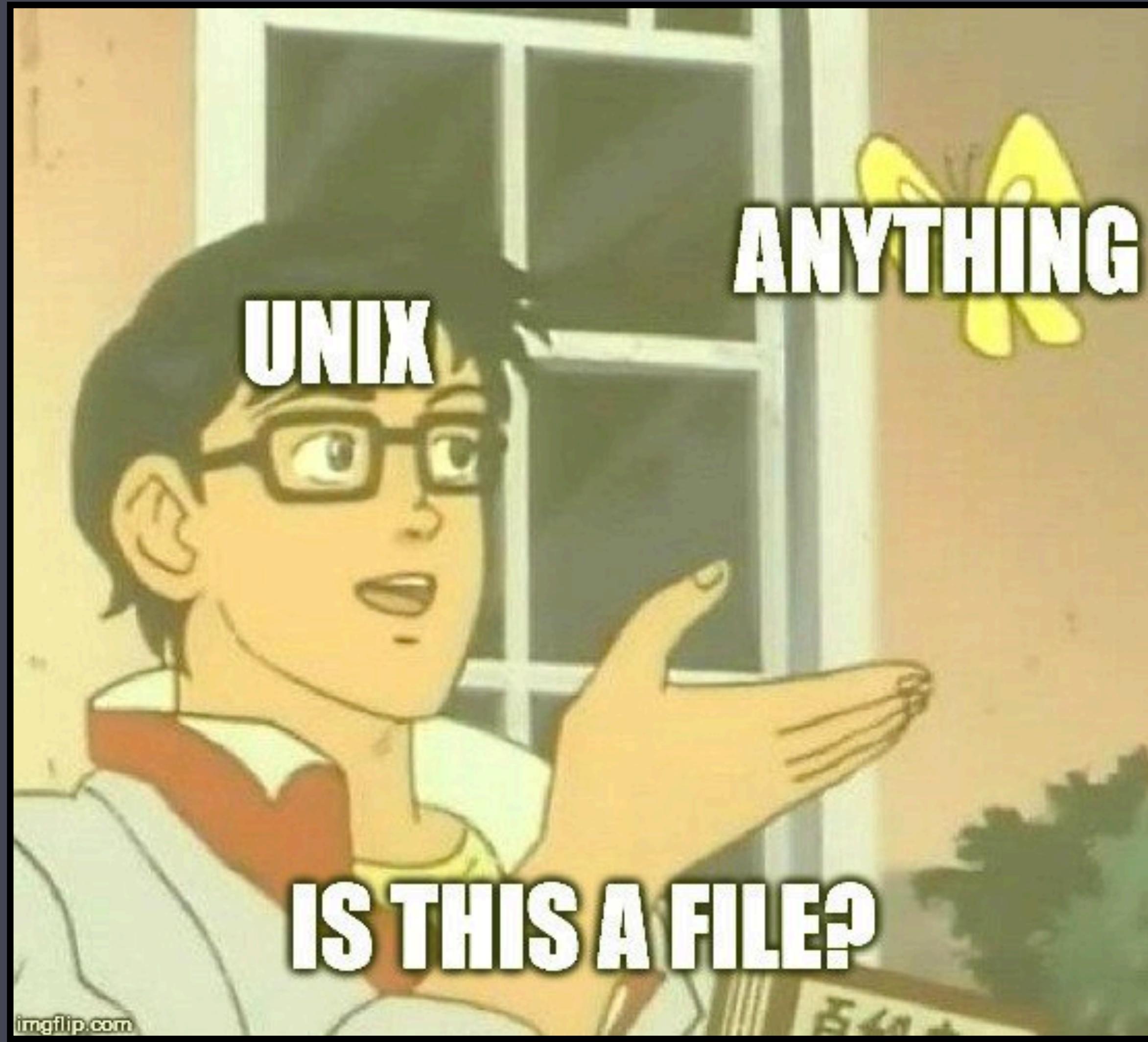
3. `ls -l | sort -n -k 5`: Lista los archivos en el directorio actual con detalles, ordenados por el tamaño del archivo en orden ascendente.

4. `ps aux | grep "proceso"`: Lista todos los procesos que se están ejecutando en el sistema y filtra solo los que contienen la cadena "proceso".

5. `cat archivo.txt | sed 's/palabra1/palabra2/g'`: Muestra el contenido del archivo archivo.txt y reemplaza todas las ocurrencias de "palabra1" con "palabra2".

Estos son solo algunos ejemplos de los muchos comandos tipo pipe que puedes usar en UNIX. La **combinación de comandos con tuberías** puede ser muy poderosa y te permite realizar operaciones más complejas y avanzadas.





WWW DEV SYS DOC VBOX CHAT MUS VID GFX xmonad.hs (~/.xmonad) - VIM

Just memorize these fourteen contextually dependant instructions

**HOTTEST EDITORS**

- 1995 — [EMACS-VIM]
- 2000 — [EDITOR WAR]
- 2005 — VIM
- 2010 — NOTEPAD ++
- 2015 — SUBLIME TEXT
- 2020 — CRISPR
- 2025 — CRISPR (VIM KEYBINDINGS)

**Exiting Vim**

Eventually

@ThePracticalDev

**NANO**  
mostly used by people who do not know what they are doing; or psychopaths.

**i3wm** vs **Qtile**

**Notepad**

**Pico**

**VIM**  
usable in just about any environment.  
does one thing, well.

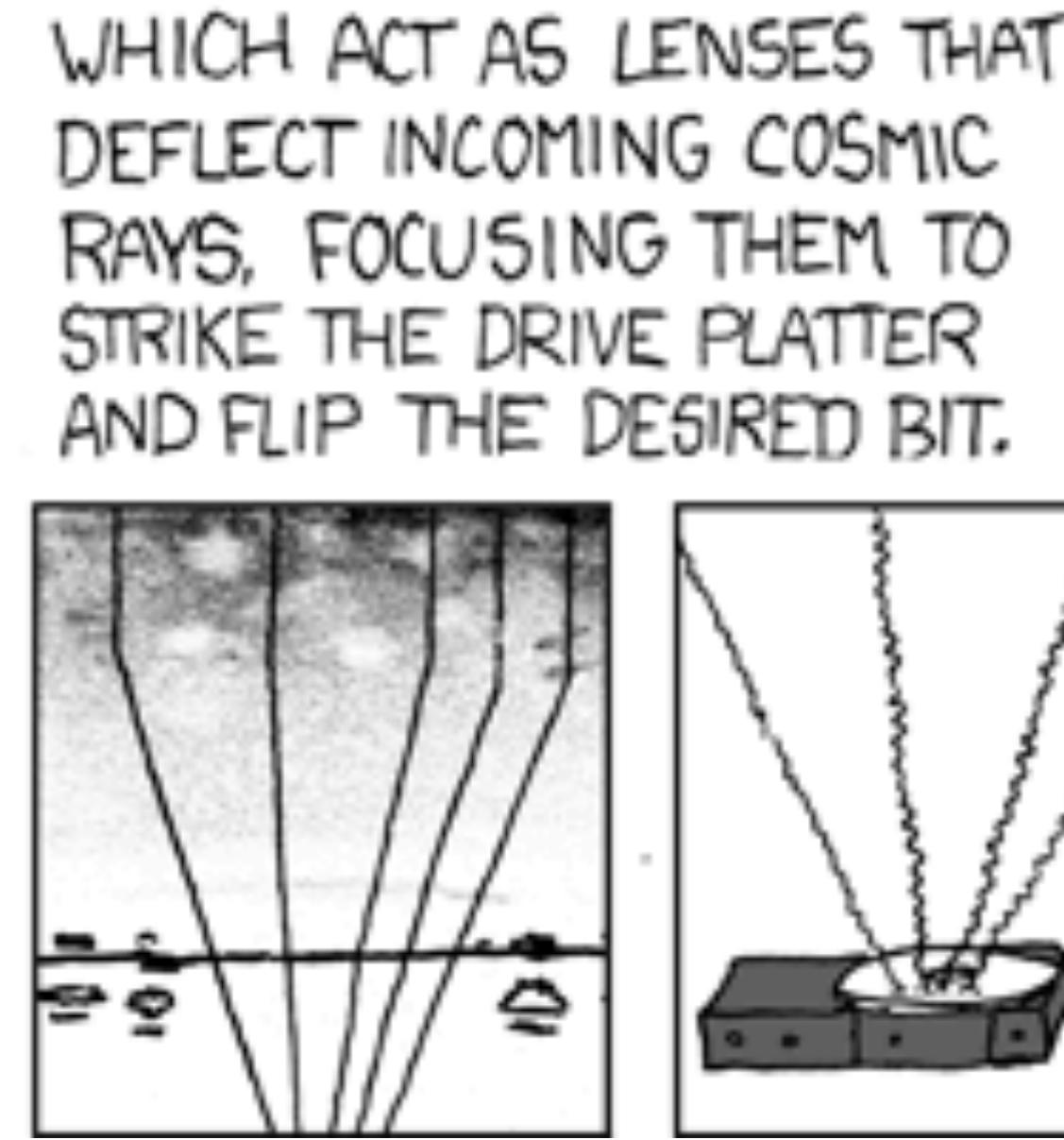
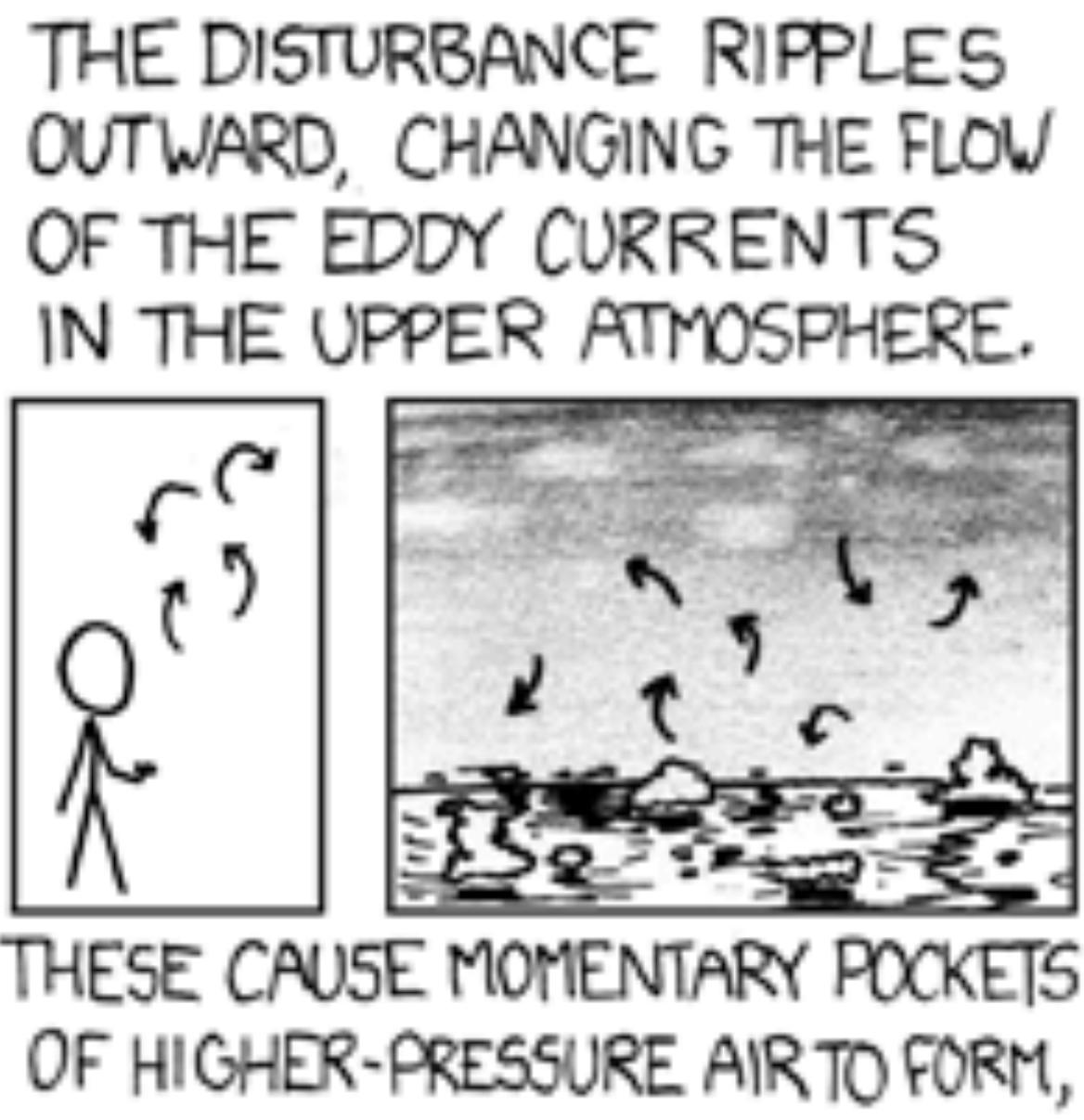
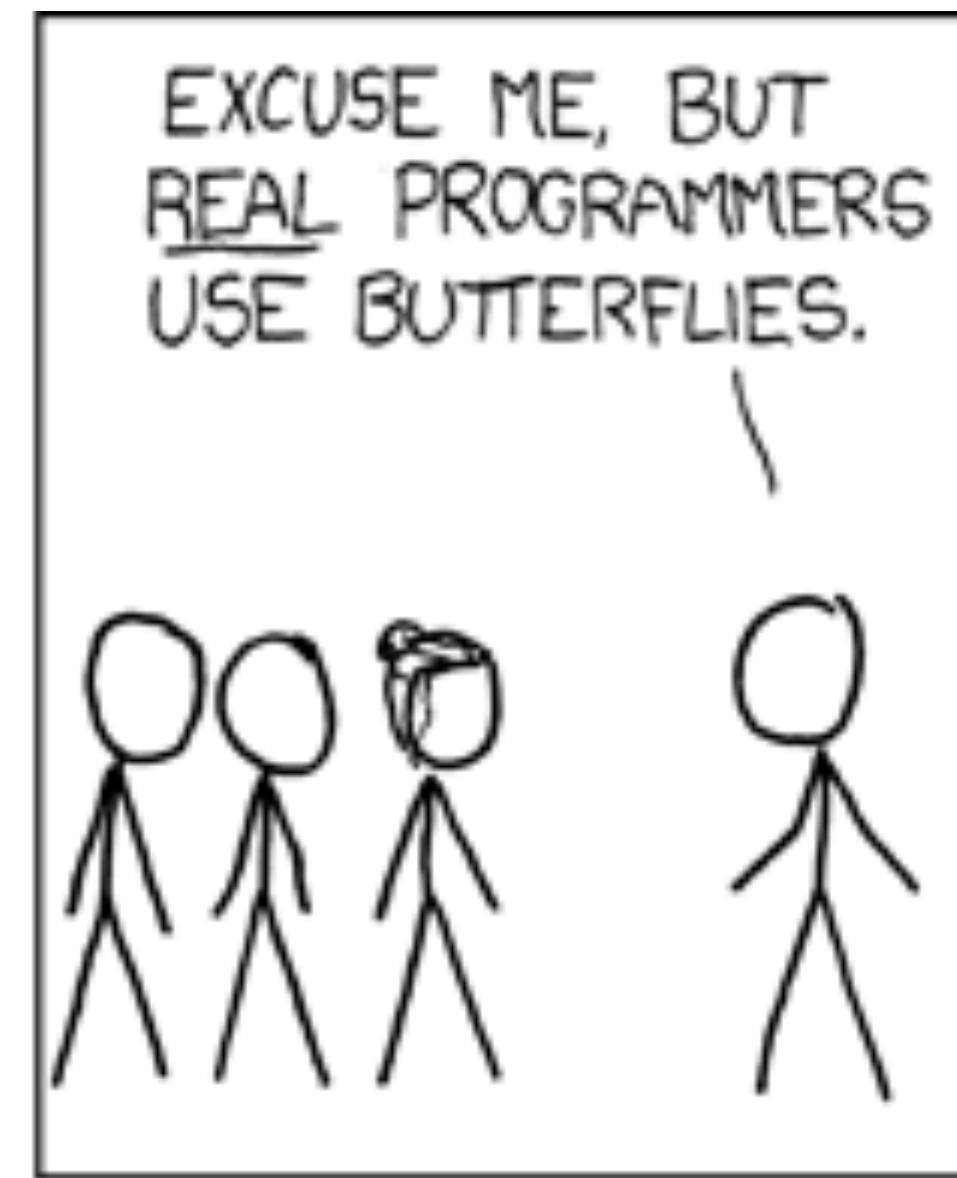
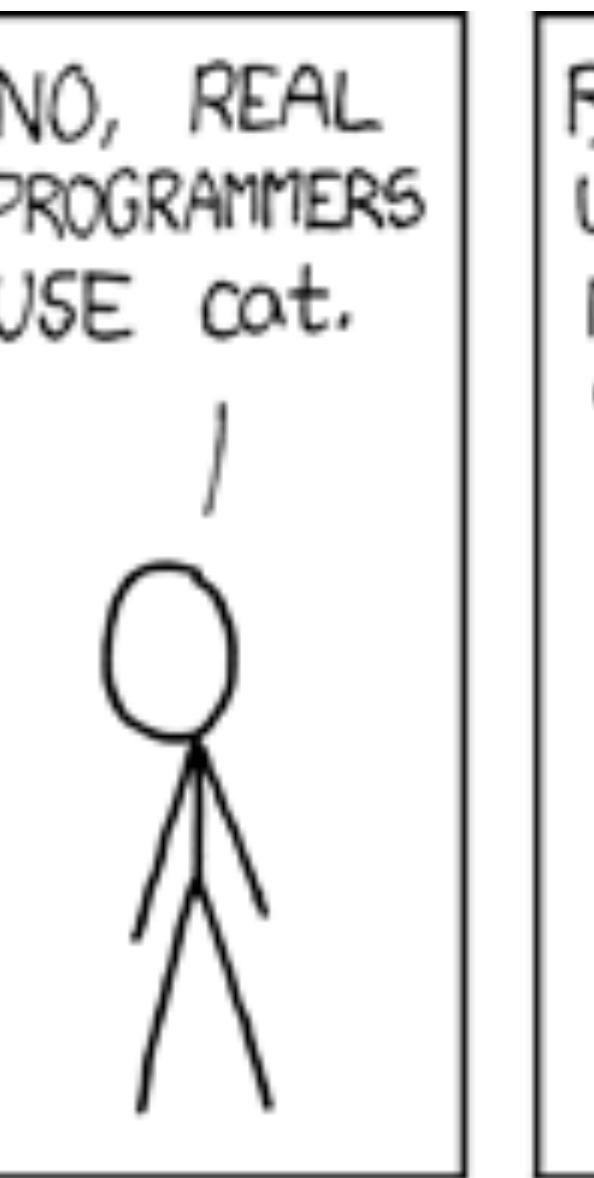
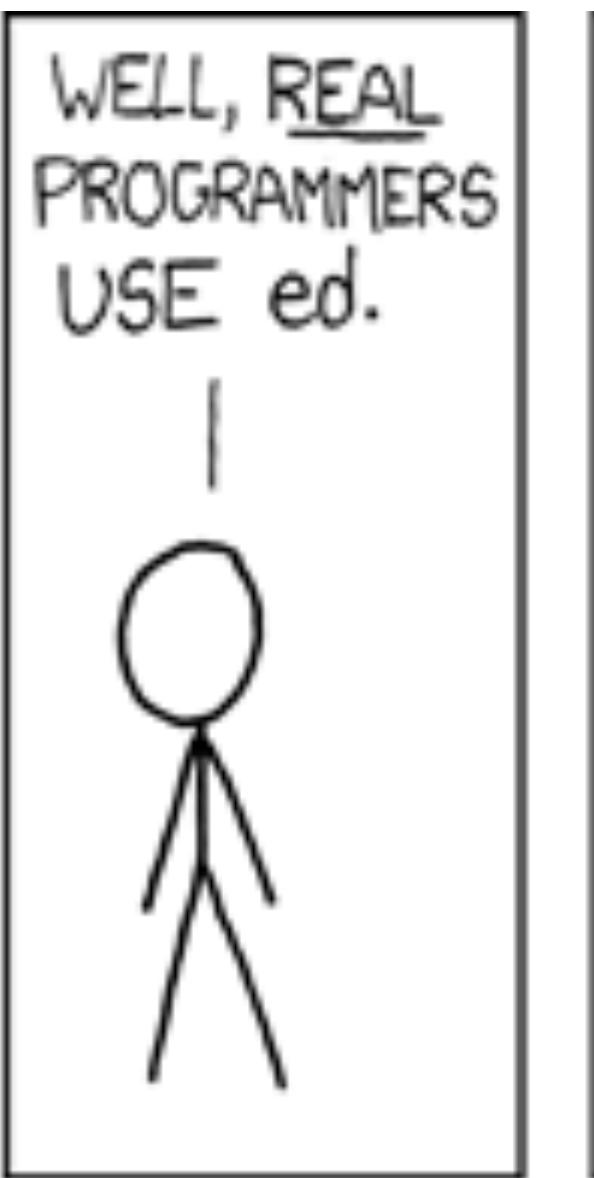
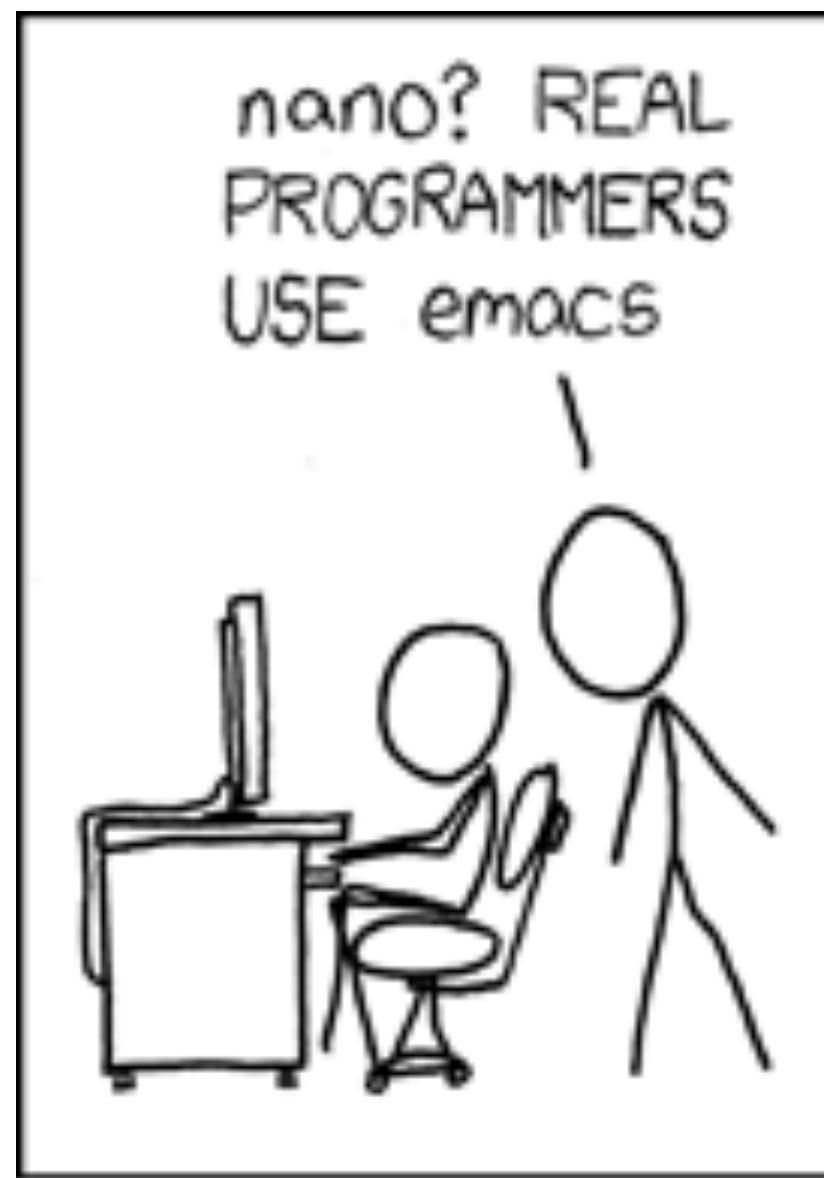
**ORLY?**

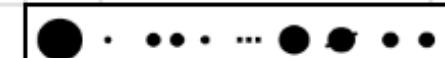
**Visual Studio**

**vi**

**emacs**

© 2015 CURTIS LASSAM - CUBE-DRONE.COM





## PROGRAMA DESARROLLO DE SOFTWARE EN ASTROFÍSICA 2023-01 | horarios 13:45

Sala 763 Pabellón Forma, Profesor Sebastián Pérez

Nº	MES	DIA	HORA	PROFESOR	TEMA	OBSERVACIONES	ESTADO
1	Marzo	14/03/2023	13:45 - 15:15	Sebastián Pérez	Introducción, Data challenge #1	<b>Programa general, evaluaciones, objetivos del curso</b> <b>Solución del data challenge #1</b> <b>Día de la Astronomía</b> <b>Cancelada por profe enfermo</b> <b>Terminal, comandos básicos, man pages, editores</b> <b>Cristóbal Donoso (UdeC, Data Science Unit)</b>	Clase normal Realizada ▾
2		16/03/2023		Sebastián Pérez	Data challenge #1 (práctico)		Clase normal Realizada ▾
3		21/03/2023		Sebastián Pérez			Otro ▾
4		23/03/2023		Sebastián Pérez			Otro ▾
5		28/03/2023		Sebastián Pérez	Planificación, Introducción UNIX		Clase normal Pendiente ▾
6		30/03/2023		Sebastián Pérez	Clase invitada (Astro data science)		Clase normal Pendiente ▾
7	Abril	4/04/2023		Sebastián Pérez	Introducción Python	<b>Programas básicos</b> <b>Programas básicos</b> <b>Experiencia de programación en hidrodinámica</b> <b>Clase práctica para aprender FARGO3D</b> <b>Buscar temas para los trabajos de desarrollo</b> <b>Versionamiento, repositorios</b> <b>Aprenderemos a hacer pull/push/commit</b> <b>Decidir sobre temas y grupos</b>	Clase normal Pendiente ▾
8		6/04/2023		Sebastián Pérez	Introducción Python		Otro ▾
9		11/04/2023		Philipp Weber (CIRAS/YEMS)	Clase invitada (FARGO 3D)		Clase normal Pendiente ▾
10		13/04/2023		Philipp Weber (CIRAS/YEMS)	Clase invitada (FARGO 3D)		Clase normal Pendiente ▾
11		18/04/2023		Sebastián Pérez	Temas de investigación		Desarrollo libre ▾
12		20/04/2023		Sebastián Pérez	Introducción a git		Clase normal Pendiente ▾
13		25/04/2023		Sebastián Pérez	Uso básico de GitHub		Clase normal Pendiente ▾
14		27/04/2023		Sebastián Pérez	Temas de investigación		Desarrollo libre ▾
15	Mayo	2/05/2023		Sebastián Pérez		<b>Primera evaluación</b>	Evaluación ▾
16		5/05/2023		Sebastián Pérez			Clase normal Pendiente ▾
17		16/05/2023		Receso USACH			Otro ▾
18		18/05/2023					Otro ▾
19		23/05/2023		Sebastián Pérez			Clase normal Pendiente ▾
20		25/05/2023		Sebastián Pérez			Clase normal Pendiente ▾
21		30/05/2023		Sebastián Pérez			Clase normal Pendiente ▾
22	Junio	1/06/2023	13:45 - 15:15	Sebastián Pérez		<b>Segunda evaluación</b>  <b>Experiencia en astroinformática</b> <b>paquetes python para big data, dask, zarr, cupy y numba</b>	Evaluación ▾
23		6/06/2023		Sebastián Pérez			Clase normal Pendiente ▾
24		8/06/2023		Miguel Cárcamo (CIRAS/YEMS)	Clase invitada Astroinformática		Clase normal Pendiente ▾
25		13/06/2023		Miguel Cárcamo (CIRAS/YEMS)	Clase invitada Astroinformática		Clase normal Pendiente ▾
26		15/06/2023		Sebastián Pérez			Clase normal Pendiente ▾
27		20/06/2023		Sebastián Pérez			Clase normal Pendiente ▾
28		22/06/2023		Sebastián Pérez			Clase normal Pendiente ▾
29		27/06/2023		Sebastián Pérez			Clase normal Pendiente ▾
30		29/06/2023		Sebastián Pérez			Evaluación final ▾
31	Julio	4/07/2023		Sebastián Pérez			Clase normal Pendiente ▾
32		6/07/2023		Sebastián Pérez			Clase normal Pendiente ▾
33		11/07/2023		Sebastián Pérez			Clase normal Pendiente ▾
34		13/07/2023		Sebastián Pérez			Clase normal Pendiente ▾
							Clase normal Pendiente ▾