

# ARQUITECTURAS DE ORDENADORES

IVÁN GONZÁLEZ MARTÍNEZ

ivan.gonzalez@uam.es

## PRÁCTICAS

JORGE LÓPEZ-DE-VERGARA

C-224

jorge.lopez-vergara@uam.es

## COORDINADORES

→ TEORÍA: francisco.gomez@uam.es

→ PRÁCTICAS: gustavo.sutter@uam.es

## CALIFICACIÓN:

$$*NT = \text{Max}([0'25P1 + 0'25P2 + 0'5\text{Final}], \text{Final})$$

⊗ los parciales no quitan temario del final

$$*NP = 0'15P1 + 0'2P2 + 0'25\text{Exp1P2} + 0'125P3 + 0'125P4 + 0'15\text{Exp3P4}$$

⊗ Mínimo un 3'5 en todas.

$$\text{NOTA FINAL} = 0'7NT + 0'3NP$$

## BIBLIOGRAFÍA

Estructura y diseño de computadores: La interfaz software/hardware

Patterson y Hennessy. Ed Reverte 2011

ISBN: 978-84-291-2620-4



# Un poco de Shell Scripting

## Navegación y exploración de archivos

- `cd` - change directory
- `ls` - list directory contents
- `find` : sirve para buscar ficheros. Comando que desciende recursivamente desde el fichero actual buscando archivos que se correspondan con una expresión dada.  
Ejemplo: `find -name "Documentación P2.pdf"`

## Visualización de ficheros

- `cat` : imprime por pantalla el contenido de un fichero
- `head` : imprime por pantalla las primeras líneas de un fichero (5 por defecto). Ejemplo: `head -n 10 <nombre-fichero>`
- `tail` : imprime por pantalla las últimas líneas de un fichero. Análogo a `head`.
- `less` : visualizador y navegador de archivos. Escribiendo una barra invertida ("`/`") seguida de la expresión a buscar y presionando "`Enter`" podemos buscar en el documento.

## Gestión de procesos y sistema

- `top` : muestra de forma dinámica información sobre los procesos en ejecución (cpu que consumen, memoria,...).
- `ps` : process status  
Muestra información sobre procesos que son controlados por una terminal. `-a` : procesos de otros usuarios  
`-l` : más información de cada proceso
- `pstree` : Muestra todos los procesos siguiendo una estructura de árbol.
- `df` : muestra la cantidad de espacio libre en las distintas particiones de los discos duros.

- du: muestra el tamaño en el directorio
- free: muestra información sobre el consumo de memoria.

## Trabajo con ficheros

- grep: busca un patrón (expresión regular) en un fichero.
- wc: word count  
cuenta las letras, palabras y líneas de un fichero.
- sort: ordena las líneas de un fichero. Existen varias opciones.
- uniq: devuelve las líneas no repetidas de un fichero. Sólo filtra las líneas repetidas consecutivas con lo que si queremos filtrar todas las líneas repetidas es necesario primero pasar el archivo por sort: `sort file | uniq`

## Redirección

- & : provoca que el comando se ejecute en segundo plano.
- | : Tubería. Redirige la salida del comando anterior al pipe como entrada al comando posterior.  
Ejemplo: `sort file | uniq | wc -l` → ordena + líneas únicas + cuenta las líneas
- > : Redirige la salida, que por defecto es stdout, al fichero deseado. Ejemplo: `sort file | uniq > file2`
- < : Redirige la entrada, que por defecto es stdin, desde el fichero deseado



