

Examen Sistemas Informáticos - Extraordinaria

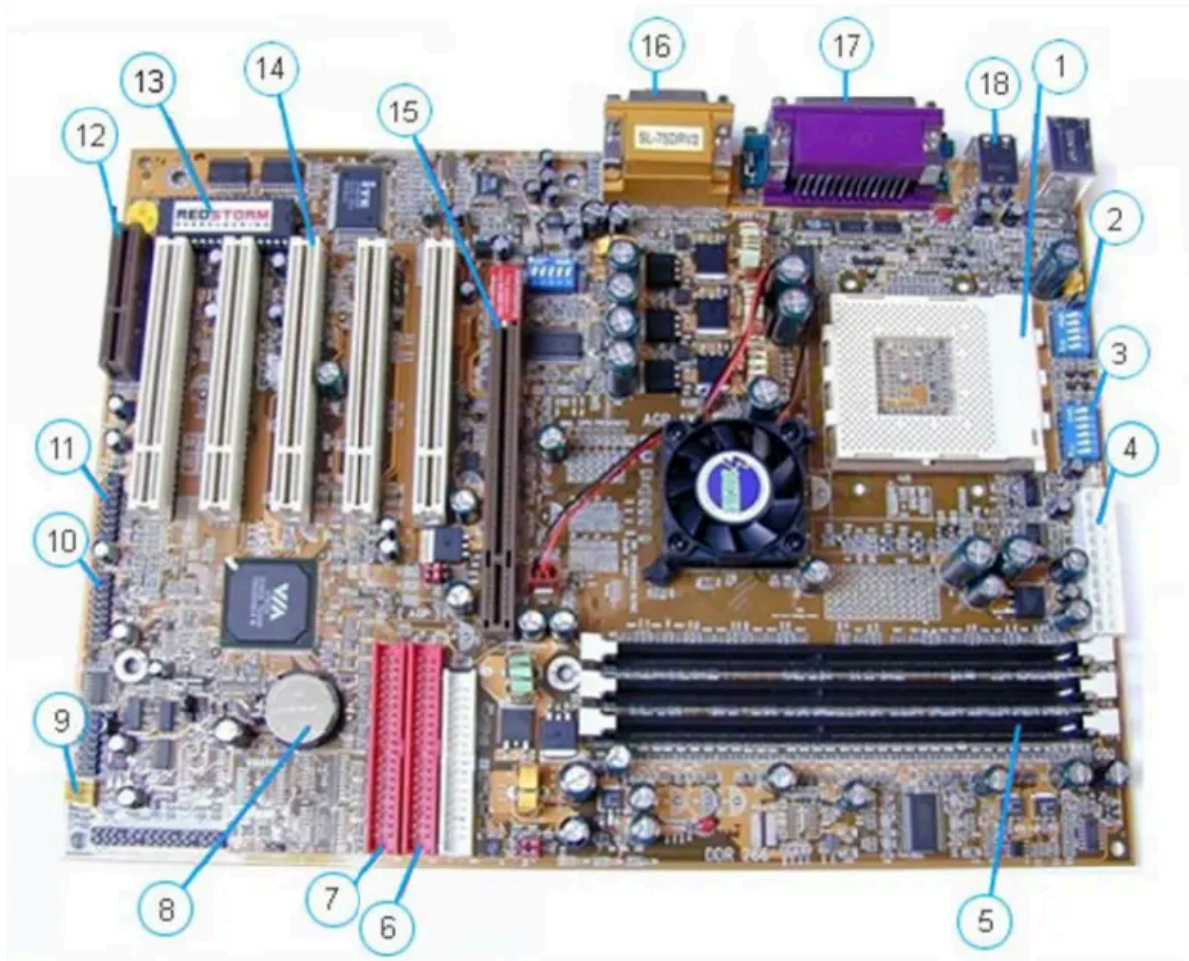
Nombre:

Apellidos:

Evaluacion 1

Hardware

1. Convierte los siguientes números de decimal a binario (1 punto)
 - a. 128
 - b. 35
 - c. 46
 - d. 322
2. De decimal a hexadecimal (1 punto)
 - a. 345
 - b. 123
 - c. 52
 - d. 67
3. Realiza un esquema de la arquitectura de Von Neuman (1 puntos)
4. Define programa y cpu atendiendo a esta arquitectura (1 puntos)
5. Identifica los distintos componentes de la siguiente imagen. (0.25 componente)
 - a. Procesador.
 - b. Conectores de alimentación.
 - c. Zocalos de RAM.
 - d. Conectores disco duro.
 - e. Conector salida de video
 - f. BIOS
 - g. Puerto PCI
 - h. SATA



6. Indica la función de los siguientes componentes. (0.5 punto /componente)
 - a. Placa base
 - b. Procesador
7. ¿Qué función tiene un sistema de archivos? ¿Qué objetivo tiene? Nombra 3 sistemas de archivos (2 puntos)
8. Identifica el sistema de archivos de tu máquina. Adjunta una captura. (1 punto)

Evaluación 2

HW2

1. ¿Que es la memoria de un ordenador? ¿Qué tipos hay? ¿Qué operaciones permite realizar? **(1,5 puntos)**
2. Define cada uno de los tipos de memoria e indica sus características. **(2 puntos)**
3. Que especifica la frecuencia, el ancho del bus y la velocidad de transferencia.**(1 punto)**
4. Define. **(3 puntos)**
 - a. *Puerto Serie*
 - b. *Puerto paralelo*
 - c. *Puerto USB*
 - d. *Puerto Thunderbolt*
 - e. *VGA*
 - f. *HDMI*
5. Cuando se hace el formato físico, indica las partes en las que se estructura un disco duro. Y el formato lógico. **(1.5 punto)**
6. Indica la estructura interna de un disco SSD y define cada una de sus partes **(1 punto)**

Redes

1. Explica la diferencia entre topología lógica y topología física. **(1,5 puntos)**
2. Define y especifica las ventajas y desventajas de las siguientes topologías: **(1 punto)**
 - a. *Bus*
 - b. *Estrella*
 - c. *Anillo*
 - d. *Malla*
3. Dibuja un esquema del modelo OSI y define brevemente cada una de sus capas **(2 puntos)**
4. Calcula la máscara de subred de las siguientes direcciones IPv4, indica a qué dirección IP pertenece, a qué clase pertenece dicha dirección IP e indica el número de dispositivos que puedes instalar en cada subred. **(2 puntos)**
 - a. *192.168.0.0*
 - b. *191.167.168.0*
 - c. *172.30.40.0*
 - d. *9.9.9.0*
5. Suponemos que tenemos la red *200.200.100.0* y queremos montar **5 subredes**. Calcula todas las **direcciones de subred**, los rangos IP de cada una de las subredes y las direcciones de Broadcast. **(3.5 puntos)**

Evaluación 3

Linux

Descarga la ISO e instala, mediante una máquina virtual, **UBUNTU**.

Una vez lo tengas instalado (**Importante:** pon tu nombre como usuario, no un alias), realiza los siguientes ejercicios en la terminal.

- 1) Crea el fichero a.txt con gedit, b.txt con nano y c.txt con vi. Guárdalos en tu subdirectorio personal. Como contenido de los ficheros introduce tu nombre, tu primer apellido y tu segundo apellido, respectivamente. Después mira el contenido de los 3 ficheros en una única línea de comandos.
- 2) Cambia la contraseña al usuario root. Entra como root. Mira como root el contenido de los ficheros /etc/shadow y /etc/gshadow de forma paginada por pantalla. Mira el contenido del fichero /etc/sudoers, y comprueba quién tiene o puede tener todos los privilegios en el sistema.
- 3) Desde la terminal, crea un grupo nuevo, llamado Profesores. Después añade un usuario llamado Alejandro, cuyo grupo primario o principal sea Profesores. Después comprueba lo realizado.
Crea otro grupo llamado Tutores. Haz que Alejandro pertenezca a Tutores, pero como grupo secundario. Comprueba lo realizado y después borra los grupos y al usuario.
- 4) Ejecuta nano d.txt. Páralo y mándalo al segundo plano. Comprueba que está en segundo plano. Vuelve a poner activo mandando al primer plano el primer proceso. Mata el segundo proceso. Comprueba que ya no hay procesos asociados a la terminal.
- 5) Crea dos usuarios nuevos, Damian y Valentin.
- 6) Cambia el Shell de Valentín por el de /bin/false. Intenta entrar en el sistema como Valentin. ¿A qué es debido que no puedas entrar? Cambia de nuevo a /bin/bash.
- 7) Crea 2 ficheros:
 - a. F.txt que contiene
ESTO
ESTA
CLARO
 - b. G.txt que contiene
CLARO
OSCURO
33

Mezclalos en un fichero fj.txt, muéstralo por pantalla y evita las líneas repetidas.

Crea otro grupo llamado Tutores. Haz que Alejandro pertenezca a Tutores, pero como grupo secundario. Comprueba lo realizado y después borra los grupos y al usuario.

- 8) ¿Dónde y qué mirarías para saber que el usuario “Damian” puede realizar comandos con privilegios de root? Dale los permisos de root si no los tiene ya
- 9) Crea una variable que almacene el resultado de 3+3 y muéstrala por pantalla
- 10) Lista todos los procesos que haya en ejecución.

- 11) Crea un fichero llamado "listado" con el listado del directorio /etc ordenado y muéstralo por pantalla.
- 12) Encuentra todos los archivos en tu directorio personal mayores que 5MB.
- 13) Averigua la cantidad de disco usada en total por el directorio home
- 14) Crea un directorio EXAMEN en /tmp y en su interior crea los directorios dir1 y en su interior dir11, dir 2 con dir21 y dir22y dir3 con dir31. Dir 31 tendra a su vez dir311 y dir312 en su interior.
- 15) Crea un script que guarde "Will you be our last ray of hope? No" en una variable y la imprima por pantalla.
- 16) Crea una variable llamada DAÑO que tenga el valor 9999. Imprime por pantalla ¿Has podido hacerlo? ¿Qué cambios se te ocurren para que funcione?
- 17) Ahora, crea otra variable llamada Vacaciones que almacene el mensaje "Ya queda menos" e imprímela por pantalla.

Automatización

1. Dado el siguiente script, indica la salida del comando. (1 punto)

```
```.binbash.sh
#!/bin/bash
```

```
if [[$1 -gt $2]]
then
 echo "$2"
elif [[$1 -lt $2]]
then
 echo "$3"
else
 echo "$1"
fi
```.
```

\$./binbash.sh 5 5 3 4

- a. La salida es 3
- b. La salida es 4
- c. La salida es 5
- d. No hay salida

2. Dado el siguiente script, indica la salida del comando. (1 punto)

```
```.binbash2.sh
```

`#!/bin/bash`

```
VAR=0
until [[$VAR -gt $2]]
do
 ((VAR++))
done
echo VAR
````
```

\$./binbash2.sh 2 5

- a. La salida es VAR**
- b. La salida es 0**
- c. La salida es 3**
- d. La salida es 6**

3. Dado el siguiente script, indica la salida del comando (1 punto).

Aclaración: “_” es lo mismo que “*imprime linea vacia*”

```
``binbash3.sh
#!/bin/bash
```

```
suma(){
    local resultado=$(( $1+$2 ))
    echo $resultado
}
echo $(suma $1 $2)
echo $resultado
````
```

**\$ ./binbash3.sh 3 4**

- a. La salida es: 7**
- b. La salida es: 7**  
**7**
- c. La salida es: 7**
- d. No hay salida** \_

**4. Crea un script llamado echo.sh que permita mostrar el argumento indicado y si se introduce un fichero deberá escribirlo en su interior. El script deberá recibir dos parámetros una palabra y fichero como parámetros y deberá dar un error si el fichero no existe o el fichero no tiene permisos de escritura. (2 puntos)**

**Ejemplo de ejecución:**

```
$./echo.sh hola
hola
```

```
$./echo.sh suspenso /home/profe/aprobados.txt
suspenso añadido a /home/profe/aprobados.txt
```

```
$./echo.sh /no/tiene/permisos/deEscritura.txt
*Error, el fichero proporcionado no tiene permisos de escritura
```

5. Crea un script llamado multitask.sh que reciba una opción y una ruta como parámetros. Si la opción es **--create** o **-c** deberá crear un fichero con el nombre indicado, si la opción es **--remove** o **-r** deberá borrar el fichero indicado y si la opción es **--append** o **-a** deberá pedir al usuario que escriba algo por pantalla y será agregado al final del fichero. **(3 puntos)**

Realiza cada una de las opciones usando una función. **(1 punto)**

**Punto extra:** Ten en cuenta que un fichero puede no existir o puede que no tengas permisos para editar dicho fichero, ten en cuenta dichos casos e implementa las funciones pertinentes. **(hasta +1 punto)**

**Ejemplo de uso:**

```
$./multitask.sh -c /home/profe/examen.txt
Fichero creado correctamente
```

```
$./multitask.sh --append /home/profe/examen.txt
¿Qué quieres añadir?
> Pregunta 5...
```

Se ha añadido correctamente.

```
$./multitask.sh --append /home/profe/examen.txt
¿Que quieres añadir?
> Pregunta 6...
```

Se ha añadido correctamente.

```
$ cat /home/profe/examen.txt
Pregunta 5...
Pregunta 6...
```

```
$./multitask.sh -r /home/profe/examen.txt
Fichero borrado correctamente
```



6. Crea un script llamado **restos.sh** que pida al usuario un número y muestre por pantalla los restos de dicho número de dividir el número entre un número entre 1 y dicho número. Se ha de suponer que el usuario va a introducir un número válido en todos los casos.

### Ejemplo de uso:

\$ ./restos.sh

Introduce un numero: 15

15 / 1; resto=0

15 / 2; resto=1

15 / 3; resto=0

.

.

.

15 / 14; resto=1

15 / 15; resto=0