

1º de Grado en Ingeniería Informática – SÓLO 2º PARCIAL
Final de enero de Fundamentos de Computadores (temas 4, 5 y 6)
22 de enero 2019

Apellidos: _____

Grupo: _____

Nombre: _____

DNI: _____

Instrucciones para realizar el examen (tipo A)

- El tiempo disponible total es de 2 horas.
- No olvide poner los apellidos y el nombre tanto en la hoja de examen como en los folios entregados.
- Para las preguntas tipo test, seleccione una única respuesta en cada cuestión en el lugar habilitado para ello (señalando con una X en la tabla colocada al comienzo del test).
- En el test, cada dos respuestas incorrectas anulan una correcta.
- Una pregunta sin contestar ni suma ni resta.

Parte I: tipo test (27%; 0.15 puntos por respuesta)

A	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18
a																		
b																		
c																		

T1. En el intérprete de comandos `bash` sobre Linux, la orden que sirve para mostrar un listado de los comandos más recientes tecleados por el usuario hasta ese momento es:

- a) `orders`
- b) `history`
- c) `commands`

T2. La orden `pwd` sirve para:

- a) Mostrar la ruta completa hasta nuestro directorio actual.
- b) Mostrar la lista de todos los procesos activos en el sistema.
- c) Mostrar la lista de procesos con permisos leer/escribir en nuestro directorio de trabajo.

T3. ¿Cuál de las siguientes órdenes listaría todos los archivos del directorio actual cuyo nombre contuviese siempre una ‘a’ mayúscula o minúscula como segundo carácter?

- a) `ls *Aa*`
- b) `ls ?[aA]*`
- c) `ls -i *{aA}2*`

T4. Si ejecutásemos la orden `chmod 754 /home/alumno/midir` (sobre dicho directorio ya existente), sería cierto que:

- a) Ningún usuario que no fuese `alumno` podría crear nuevos archivos dentro de dicho directorio.
- b) El usuario `alumno` podría ejecutar cualquier archivo contenido en dicho directorio.
- c) Los usuarios pertenecientes al grupo propietario de dicho directorio podrán modificar todos los archivos contenidos en dicho directorio.

T5. El principal creador inicial de Linux, allá por el año 1991, fue un tal Linus...

- a) Kernighan.
- b) Tanenbaum.
- c) Torvalds.

T6. El *scheduler* de un sistema operativo pertenece a su subsistema de gestión de:

- a) Procesos.
- b) Memoria.
- c) Entrada/Salida.

T7. En el proceso de traducción, cuando ya hemos obtenido el código ejecutable final, la ventaja de haber usado librerías estáticas o dinámicas consiste en:

- a) Que si hemos usado librerías dinámicas, no es necesario que existan en el ordenador donde ejecutaremos ese programa.
- b) Que si hemos usado librerías estáticas, el programa ocupará menos espacio en disco y ocupará, cuando se ejecute, menos espacio en memoria de lo que ocuparía con las dinámicas.
- c) tanto a) como b) son falsas.

T8. Para desapilar el contenido de la cima de la pila en el registro `%rbx`:

- a) Utilizo la instrucción `push %rbx`.
- b) Utilizo la instrucción `ret %rbx`.
- c) Utilizo la instrucción `pop %rbx`.

T9. El módulo objeto es un fichero:

- a) De instrucciones en ensamblador (legible con un editor de texto ASCII) pero faltando el proceso de enlazado/*linkado*.
- b) De instrucciones binarias parcialmente traducidas a lenguaje máquina, pero con necesidad aún de enlazado/*linkado*.
- c) Ejecutable final, objeto del programa que hemos realizado con un lenguaje de alto nivel y después compilado y enlazado/*linkado*.

(Sigue por detrás)

T10. Una característica del ISA x86_64 de Intel es que:

- a) Todas sus instrucciones no tienen el mismo tamaño.
- b) El resultado de una subrutina es devuelto al código invocador mediante la pila.
- c) Los saltos incondicionales utilizan el registro de estado establecido por una instrucción anterior.

T11. Las operaciones de desplazamiento de bits en x86-64 pertenecen a la categoría de instrucciones:

- a) Aritmético-lógicas.
- b) De movimiento de datos.
- c) De soporte para procedimientos.

T12. El formato de la codificación de una instrucción en x86-64 trata de:

- a) Cómo se distribuyen los bits para el código de la operación, operandos fuente y destino, etc.
- b) Si es aritmético-lógica, de acceso a memoria, de saltos, etc.
- c) Si es de tipo RISC o de tipo CISC.

T13. Dada la red global, 6.7.8.0/23, ¿cuántas subredes diferentes puedo llegar a obtener si se definen subredes con máscara 255.255.255.192?

- a) 8.
- b) 16.
- c) 32.

T14. El protocolo DHCP permite:

- a) Recibir/enviar correo a través de un servidor de correo electrónico.
- b) Dada una dirección Web de un servidor, obtener la correspondiente dirección IP.
- c) Obtener la dirección IP del equipo, su máscara de red, y la dirección IP del *router* por defecto.

T15. La aplicación *BitTorrent* utiliza como modelo de comunicación:

- a) Un modelo cliente-servidor.
- b) Un modelo peer-to-peer.
- c) Un modelo servidor-servidor.

T16. IPv6 soluciona uno de los siguientes problemas de IPv4:

- a) La carencia de direcciones IP públicas.
- b) Elimina la necesidad de las tablas de enrutamiento y, por tanto, no será necesario usar *routers*.
- c) Elimina la necesidad de usar máscaras de red.

T17. Una de las siguientes afirmaciones es **FALSA**. El uso de puertos en la capa de transporte permite:

- a) Que diferentes procesos atiendan servicios de red diferentes en un mismo servidor.
- b) Que un mismo host sea accesible a través de dos o más direcciones IP.
- c) Que se use el truco del NAT para reutilizar direcciones IP privadas y aliviar así la escasez de direcciones IP versión 4.

T18. El protocolo IMAP:

- a) Permite dividir un rango de direcciones IP en distintas subredes de forma automática.
- b) Controla y maneja los posibles errores de conexión a un servidor.
- c) Permite recibir correo electrónico.

1º de Grado en Ingeniería Informática – SEGUNDO PARCIAL

Segundo parcial de Fundamentos de Computadores (temas 4, 5 y 6)

22 de enero de 2019

Apellidos: _____

Grupo: _____

Nombre: _____

DNI: _____

Instrucciones para realizar el examen

- El tiempo total disponible es de 2 horas (desde el inicio del examen).
- Las preguntas se contestarán en folios.
- No olvide poner los apellidos y el nombre tanto en la hoja de examen como en los folios entregados.
- Entregad tanto el enunciado del examen como los folios utilizados o no al acabar el examen.

Parte II: cuestiones teórico-prácticas (36%; puntuación indicada en cada apartado)

C1. (1.2 puntos) Contestar muy brevemente a las siguientes cuestiones:

- Explicar brevemente cuáles son los elementos tanto hardware como software que gestionan la E/S en un sistema operativo (dos frases).
- Explicar brevemente cómo el sistema operativo permite compartir la CPU entre dos procesos A y B, y qué ocurre cuando uno de ellos quiere realizar una operación de E/S.
- Explicar la diferencia entre excepción e interrupción hardware. Poner un ejemplo de ambas.

C2. (1.2 puntos) Contestar concisamente a las siguientes cuestiones:

- Explicar muy brevemente (una frase por registro) el cometido fundamental de los registros contador de programa (PC) y registro de instrucción (RI) en una CPU.
- ¿Cómo se llama exactamente el PC en la arquitectura Intel x86-64? ¿Y los registros base y cima de la pila, respectivamente? ¿Qué longitud en bits tiene cada uno de ellos?
- En relación al número y complejidad de instrucciones que puede manejar una determinada CPU, se habla de computadores de dos tipos fundamentales de tendencias: una que tiende a implementar mayor número de instrucciones, y con más variedad, y otra que hace todo lo contrario (menos instrucciones, menor variedad). Indica las siglas con las que se conoce a cada una de esas tendencias, así como el significado (en inglés) de las mismas.
- Indicar en qué dos segmentos se divide un programa escrito en ensamblador de x86-64, y qué directivas de ensamblador sirven para marcar el comienzo de sendas partes.
- Para el ISA x86-64, poner un ejemplo concreto completo (es decir, que pudiera ser una línea completa de un programa escrito en ensamblador) de (pueden inventarse las etiquetas allí donde hagan falta):
 - Una instrucción aritmético lógica que use una constante.
 - Una instrucción de llamada a una subrutina.
 - Un salto condicional a una etiqueta de código creada por el compilador.
 - Una instrucción de transferencia de datos entre dos registros de 64 bits.

e5) Una instrucción que pudiera usarse para escribir en una posición de un array.

e6) La última instrucción de una subrutina.

C3 (1.2 puntos) El Protocolo que permite la asignación automatizada (no manual) de los valores de configuración de un equipo en red se denomina **(C3A)**. Los protocolos TCP y UDP son dos protocolos de la capa o nivel de **(C3B)**. Si queremos hacer una retransmisión multimedia en tiempo real en internet lo más adecuado será hacer uso (de los dos anteriores) del protocolo: **(C3C)**. Para visualizar una página web de publicidad el protocolo de aplicación más adecuado sería el protocolo **(C3D)**, mientras que para visualizar una cuenta bancaria sería más adecuado el protocolo encriptado equivalente **(C3E)**. El servicio **(C3F)**, finalmente, es el que nos permite no tener que introducir la IP numérica del servidor en una URL.

En cuanto al correo electrónico el protocolo SMTP pertenece a capa o nivel de **(C3G)**, si queremos acceder a los mensajes a través de un programa cliente descargándolos previamente haremos uso del protocolo **(C3H)**, y si por el contrario deseamos dejarlos en el servidor haremos uso del protocolo **(C3I)**; en cualquier caso, el protocolo utilizado para el envío de mensajes al servidor propio y entre servidores es **(C3J)**.

Parte III: ejercicios boletines (37%; puntuación indicada en cada apartado)

P1. (1.3 puntos) En una línea de órdenes de Linux, se han ejecutado las siguientes instrucciones y el resultado es el mostrado:

```
[pilar@momo Fundamentos]$ ls -l
drwxrwxrwx  6  pilar profesor    4096 ene  8  12:07  Apuntes
drwxrwxr--  2  pilar profesor    4096 ene  8  12:02  Prácticas
-rw-rw-r--  1  pilar profesor   523228 ene  8  12:02  tema5.pdf
-rw-rw-r--  1  pilar profesor   49238 ene  8  12:12  boletín5.pdf
-rwxr-xr--  1  pilar profesor    3228 ene  8  12:10  analiza.sh
[pilar@momo Fundamentos]$ du -sh Apuntes/
36M    Apuntes/
```

Usando esta información, resuelva las siguientes cuestiones:

- Podrías determinar el tamaño, incluyendo su contenido, de los directorios *Apuntes* y *Prácticas*, respectivamente? Indica, en cada caso y si es posible, de dónde obtienes esta información.
- Indicar qué orden se tendría que ejecutar para mostrar por el terminal los ficheros o directorios ocultos que hay en el directorio Fundamentos.
- Indicar qué pasaría si el usuario **pilar** ejecutase la orden `cp tema5.pdf Apuntes`
- Indicar qué orden se tendría que ejecutar para crear un directorio llamado *Exámenes*.
- Indicar qué orden habría que ejecutar para mover el fichero *boletín5.pdf* al directorio *Prácticas*.
- Escribir la orden completa a ejecutar para crear un fichero comprimido llamado *Apuntes.tgz* con los contenidos de todo lo que cuelga (recursivamente) del directorio *Apuntes*.

- g) Escribir la orden a ejecutar para que el usuario **antonio, que no pertenece al grupo profesor**, pudiera ejecutar el fichero *analiza.sh*. ¿Quién o quiénes podrán ejecutar esta orden de cambio de permisos?
- h) Indicar cuál sería el efecto completo de la siguiente línea de comandos:
[pillar@momo Fundamentos]\$ find . -iname "t*" -size +1K -type f

P2. (1.2 puntos) Considérese la siguiente sesión de terminal en una máquina Linux:

```
user@host:~$ ifconfig
eth0  Link encap:Ethernet  direcciónHW 00:1d:7d:ab:0d:3e
Direc. inet:155.54.205.34  Difus.:[...] Másc:255.255.254.0
[...]
user@host:~$ nmcli
[...]
DNS configuration:
  servers: 155.54.1.10 155.54.1.1 155.54.1.2
[...]
user@host:~$ host www.wikipedia.com
www.wikipedia.com has address 91.198.174.192
www.wikipedia.com has IPv6 address 2620:0:862:ed1a::1
```

- a) Aparte de la IP propia mostrada por el primer comando (155.54.205.34), ¿qué otra IP está siempre disponible para hacer referencia al mismo host (*localhost*)? (dar dicha IP en el formato numérico habitual, es decir, X.Y.Z.W)
- b) ¿Cuántas tarjetas de red pueden conectarse como máximo a la subred a la que pertenece este host?
- c) Indicar la identificación exacta de dicha subred, en formato X.Y.Z.W/V. ¿Cuales son las direcciones de *broadcast* y de *red* de dicha subred?
- d) ¿Qué comando deberíamos utilizar para averiguar la IP de nuestro *router*? ¿Sería la IP 155.54.205.0 una dirección válida para el mismo? Razona muy brevemente la respuesta.
- e) ¿Qué protocolo utiliza el último comando (host)? Dar las siglas correspondientes, su significado, y una breve explicación de para qué sirve ¿A qué IP (o IPs) exacta(s) se conecta dicho comando, en el caso mostrado, para responder?

P3: (1.2 puntos) Dado el código en ensamblador que se muestra a continuación, responde a las siguientes preguntas:

<pre>[...] .data array: .long -4 .long 0 .long -3 .long 0 .long -2 .long 0 .long -1 .long 0 .long 1 .long 0 .long 2 .long 0 .long 3 .long 0 .long 4 .long 0 [...].text [...].main: pushq %rbp movq %rsp, %rbp subq \$16, %rsp movl \$0, -4(%rbp) jmp .L2 .L3: movl -4(%rbp), %eax cltq movl array(,%rax,4), %eax movl %eax, %edi movl \$0, %eax</pre>	<pre> call opuesto movl -4(%rbp), %edx movslq %edx, %rdx movl %eax, array(,%rdx,4) addl \$1, -4(%rbp) .L2: cmpl \$15, -4(%rbp) jle .L3 movl \$0, -4(%rbp) jmp .L4 .L5: movl -4(%rbp), %eax cltq movl array(,%rax,4), %eax movl %eax, %esi movl \$.LC0, %edi movl \$0, %eax call printf addl \$1, -4(%rbp) .L4: cmpl \$15, -4(%rbp) jle .L5 movl \$10, %edi call putchar leave ret [...].opuesto: pushq %rbp movq %rsp, %rbp movl %edi, -4(%rbp) movl -4(%rbp), %eax negl %eax popq %rbp ret [...]</pre>
---	---

- a) ¿Qué tamaño en bytes tiene la estructura de datos array?
- b) Indica todas las instrucciones de ensamblador que copian un valor de un registro a otro.
- c) Indica todas las instrucciones de ensamblador que incondicionalmente provocan que el procesador salte a una instrucción que no es la que está inmediatamente después en el código.
- d) Indica todas las instrucciones de ensamblador que condicionalmente provocan que el procesador salte a una instrucción que no es la que está inmediatamente después en el código.
- e) Describe qué hace la instrucción `movl array(,%rax,4), %eax`?
- f) Indica todas las instrucciones de ensamblador que modifican la zona de memoria perteneciente a la pila.

Soluciones a SÓLO 2º PARCIAL:

Test:

Sólo 2º parcial A

A	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18
a		X		X		X				X	X	X	X			X		
b	X		X						X						X		X	
c					X		X	X						X				X

Cuestiones:

C1: Consultar las transparencias de teoría, en concreto las secciones 4.3.3, 4.3.1 y 4.6.2.

C2:

- a) *Contador de programa (PC)*: indica en todo momento la dirección de la instrucción a ejecutar. *Registro de instrucción (RI)*: contiene la codificación máquina de la instrucción en sí a ejecutar (su código binario).
- b) El PC en la arquitectura Intel X86-64 bits se denomina `%rip`, el registro base de la pila `%rbp`, y la cima de la pila `%rsp`. Todos ellos son de 64 bits.
- c) Mayor número de instrucciones: *CISC (Complex Instruction Set Computers)*. Menor número de instrucciones: *RISC (Reduced Instruction Set Computers)*.
- d) Segmento de datos (directiva `.data`) y segmento de código (directiva `.text`).
- e) e1) `addl $1, %eax`
e2) `call funcion`
e3) `jbe .L7`
e4) `movq %rax, %rdx`
e5) `movl %eax, array(,%rsi,4)`
e6) `ret`

C3.

C3A: DHCP
C3B: Transporte
C3C: UDP
C3D: HTTP
C3E: HTTPS
C3F: DNS
C3G: Aplicación
C3H: POP
C3I: IMAP
C3J: SMTP

Ejercicios Boletines:

P1:

- a) El tamaño del directorio *Apuntes* es de 36M, lo sabemos por el resultado de la orden `du`. Sin embargo, no es posible saber el tamaño del directorio *Prácticas*.
- b) Hay que ejecutar la orden `ls -a`, ya que la opción `-a` permite visualizar los ficheros o directorios ocultos.
- c) El fichero *tema5.pdf* se copiaría en el interior del directorio *Apuntes*, pero no se borraría ya que se realiza una copia del mismo y se mantiene el original.
- d) `mkdir Exámenes`
- e) `mv boletín5.pdf Prácticas`
- f) `tar czvf Apuntes.tgz Apuntes`
- g) `chmod 755 analiza.sh`
La orden la tiene que ejecutar el usuario **pilar**, por ser la propietaria del fichero, o bien el usuario **root** ya que tiene permisos para poder ejecutar cualquier orden.
- h) Listaría todos los ficheros cuyo nombre empiece por `t` o `T`, que tengan un tamaño igual o superior a 1K y que sean de tipo fichero.

P2:

- a) 127.0.0.1 (IP que siempre referencia al propio *host*, llamado *localhost*).
- b) 510 ($2^9 - 2 = 510$, ya que la máscara de red, 255.255.254.0, acaba con 9 ceros al ser pasada a binario).
- c) 155.54.204.0/23. *Broadcast*: 155.54.205.255. *Red*: 155.54.204.0
- d) `route` (o también `nmcli`). Y sí, la IP 155.54.205.0 sería válida para un *router* en dicha subred (ya que dicha dirección está dentro de la subred, y no coincide ni con la de *red* ni con la de *broadcast*).
- e) *DNS (Domain Name Server)*: servicio que traduce IPs a nombres de dominio y viceversa. En este caso, nuestros DNS tienen IPs 155.54.1.10, 155.54.1.1, 155.54.1.2 (tal y como nos informó el comando `nmcli`), de forma que el comando `host` consultará a cualquiera de ellos para realizar la traducción del nombre de dominio `www.wikipedia.com` a su IP correspondiente, en este caso 91.198.174.192

P3.

- a) Array consiste en 16 elementos donde el tamaño de cada uno viene especificado por la directiva `.long`, que indica 32 bits (4 bytes). El tamaño es $16 * 4 = 64$ bytes.
- b) `movq %rsp, %rbp; movl %eax, %edi; movslq %edx, %rdx; movl %eax, %esi; movq %rsp, %rbp`
- c) `jmp .L2; call opuesto; jmp .L4; call printf; call putchar; ret`
- d) `jle .L3; jle .L5`
- e) Copia el contenido de la dirección de memoria indicada por `array+%rax*4` en

el registro `eax`. En realidad, dicho contenido es exactamente la posición `[i]` del array, asumiendo que el valor de la variable `i` está en ese momento en el registro `%rax`.

f) `) pushq %rbp; movl $0, -4(%rbp); call opuesto; call printf; call putchar; addl $1, -4(%rbp); movl %edi, -4(%rbp)`