

Este examen suma un total de 20 puntos. Cada 3 preguntas de test incorrectas restan 1 punto. Sólo una opción es correcta a menos que se indique algo distinto. No está permitido el uso de calculadora. La duración del examen es de 60 minutos.

Respecto a la HOJA DE RESPUESTAS:

- Rellene sus datos personales en el formulario superior.
- Indique «Redes de Computadores II» en el campo EVALUACIÓN.
- Indique su DNI en la caja lateral (número y celdillas).
- Para las preguntas que no quieras contestar, marque la opción e).

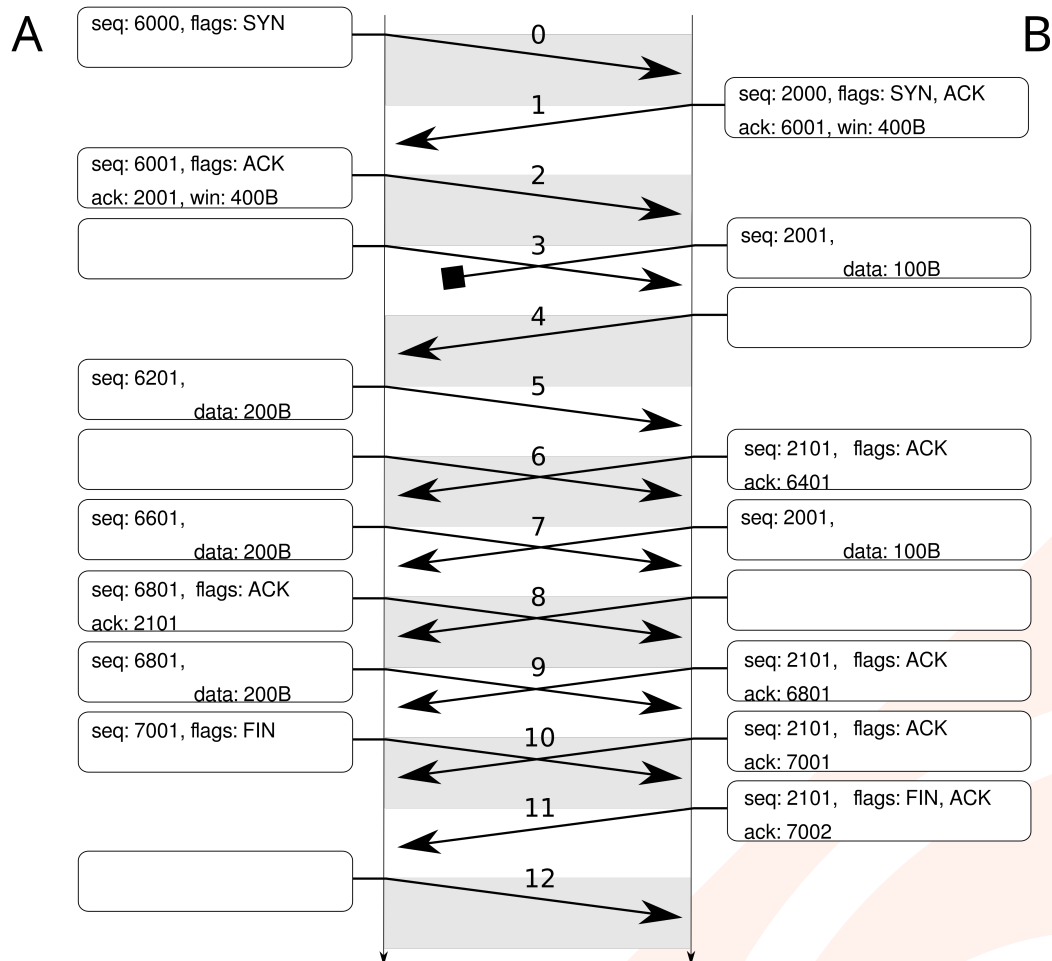
Marque sus respuestas sólo cuando esté completamente seguro. El escáner no admite correcciones ni tachones de ningún tipo, las anulará automáticamente. Debe entregar únicamente la hoja de respuestas.

- 1** [1p] ¿Cuál es la función principal del protocolo DNS en TCP/IP?
- ☐ a) Convertir nombres de dominio en direcciones IP. ☐ c) Enrutamiento de paquetes IP en la red.
- ☐ b) Encriptar la comunicación entre hosts. ☐ d) Proporcionar servicios de transferencia de archivos.
- 2** [1p] ¿Cuál es la función principal del protocolo ARP en TCP/IP?
- ☐ a) Enviar paquetes IP a través de la red local.
- ☐ b) Traducir nombres de dominio en direcciones IP.
- ☐ c) Determinar la dirección IP de un host a partir de su dirección MAC.
- ☐ d) Establecer conexiones TCP entre hosts.
- 3** [1p] ¿Cuál es el propósito del protocolo ICMP en TCP/IP?
- ☐ a) Proporcionar servicios de correo electrónico.
- ☐ b) Facilitar la resolución de direcciones IP.
- ☐ c) Establecer conexiones TCP entre hosts.
- ☐ d) Realizar pruebas de conectividad y reportar errores en la red.
- 4** [1p] ¿Cuál es la función principal del protocolo HTTP en TCP/IP?
- ☐ a) Facilitar la transferencia de información en la World Wide Web.
- ☐ b) Encriptar la comunicación entre dos hosts.
- ☐ c) Permitir que múltiples dispositivos compartan una dirección IP pública.
- ☐ d) Asignar direcciones IP y configuraciones de red automáticamente a dispositivos en una red.
- 5** [1p] ¿En el contexto de la programación de sockets en Python, ¿qué es un socket bloqueante y qué es un socket no bloqueante?
- ☐ a) Un socket bloqueante es aquel que no permite la comunicación simultánea con múltiples clientes, mientras que un socket no bloqueante permite la comunicación con varios clientes a la vez
- ☐ b) Un socket bloqueante es aquel que permite la comunicación simultánea con múltiples clientes, mientras que un socket no bloqueante solo permite la comunicación con un cliente a la vez.
- ☐ c) Un socket bloqueante es aquel que detiene el flujo del programa cuando se envían o reciben datos, esperando hasta que se complete la operación, mientras que un socket no bloqueante no detiene el flujo y permite realizar otras tareas mientras se envían o reciben datos.
- ☐ d) Un socket bloqueante es aquel que no permite la comunicación a través de Internet, mientras que un socket no bloqueante es adecuado para la transmisión de datos en redes locales.

- 6** [1p] ¿Cuál es la función del método `listen()` en un servidor TCP en Python cómo se relaciona con el método `accept()`?
- ☐ a) El método `listen()` establece el socket en modo de escucha, permitiendo que el servidor acepte conexiones entrantes. Después de llamar a `listen()`, el método `accept()` se utiliza para aceptar una conexión entrante y crear un nuevo socket dedicado para esa conexión.
 - ☐ b) El método `listen()` se utiliza para enviar datos a través de un socket TCP, mientras que el método `accept()` se utiliza para recibir datos.
 - ☐ c) El método `listen()` establece una conexión TCP con un servidor remoto, mientras que el método `accept()` se utiliza para verificar si el servidor remoto está disponible y aceptar la conexión.
 - ☐ d) El método `listen()` establece un canal seguro de comunicación entre el cliente y el servidor, mientras que el método `accept()` se utiliza para autenticar al cliente antes de establecer la conexión.
- 7** [1p] ¿Qué medida puede tomar el receptor para evitar el síndrome de la *ventana tonta*?
- ☐ a) Activar el algoritmo de Nagle.
 - ☐ b) Desactivar el algoritmo de Nagle.
 - ☐ c) No anunciar tamaños de ventana `rwnd` inferiores a un cierto umbral.
 - ☐ d) Enviar cualquier tamaño de ventana `rwnd` mayor que 0.
- 8** [1p] ¿Qué relación guardan la ventana de emisión y la ventana de recepción en el protocolo de *ventana deslizante*?
- ☐ a) La ventana de emisión no debe ser nunca más grande que la de recepción.
 - ☐ b) La ventana de emisión no debe ser nunca más pequeña que la de recepción, para que siempre haya datos disponibles.
 - ☐ c) La ventana de emisión debe ser igual que la de recepción, para que funcione el protocolo.
 - ☐ d) Es indiferente siempre que esté activado el algoritmo de Nagle.
- 9** [1p] El Timer de Retransmisión, RTO:
- ☐ a) Es imprescindible en UDP y no suele implementarse en TCP.
 - ☐ b) Es diferente en cada conexión y cambia en función del Round Trip Time.
 - ☐ c) Como mínimo tiene un valor de 60s para no saturar la red.
 - ☐ d) Todas las anteriores son verdaderas.
- 10** [1p] ¿Cuál de los siguientes NO ES un motivo por el que TCP modifica el valor del campo *sequence number* de una cabecera?
- ☐ a) Cuando el flag SYN está activo.
 - ☐ b) Cuando el segmento contiene datos.
 - ☐ c) Cuando el flag FIN está activo.
 - ☐ d) Cuando el flag ACK está activo.
- 11** [1p] Un segmento TCP atraviesa cuatro redes con MTUs=1500, 1501, 1024 y 540 bytes, respectivamente, hasta alcanzar su destino. ¿Cuál sería el valor de MSS del segmento TCP si se pretende evitar la fragmentación?
- ☐ a) 1500 bytes independientemente de todo puesto que es el valor estándar y no puede ser menor.
 - ☐ b) 500 bytes.
 - ☐ c) 1024 bytes.
 - ☐ d) Al menos 1501 bytes.
- 12** [1p] Suponga que un proceso TCP tiene una ventana de envío (`swnd`) de 200 bytes, siendo el primer byte no confirmado el 25. En este momento, se produce el envío de los datos [50:100] e inmediatamente después se recibe un `ACK = 45`. ¿Cuál sería el contenido de la ventana, y cuál sería el número de secuencia del primer byte No Confirmado (NC) y el del primer byte No Enviado (NE)?
- ☐ a) `swnd = [25,225]`, `NC=45`, `NE=100`
 - ☐ b) `swnd = [46,245]`, `NC=46`, `NE=101`
 - ☐ c) `swnd = [45,244]`, `NC=45`, `NE=101`
 - ☐ d) `swnd = [101,300]`, `NC=100`, `NE=101`

E. [5p] La figura muestra un flujo TCP en el que se utiliza control de congestión. Sabiendo que: Incluye conexión y desconexión, solo se pueden enviar segmentos coincidiendo con un tick, el plazo de retransmisión es de 4 ticks, se enviará datos siempre que se pueda y que A debe enviar 1000 bytes.

Responda a las siguientes preguntas:



- > **13** Segmento enviado por A en el tick 3:
- ☐ a) seq:6001, payload:100B
- ☐ b) seq:6001, payload:200B
- ☐ c) seq:6001, payload:100B
- ☐ d) seq:6101, payload:200B
- > **14** Segmento enviado por B en el tick 4:
- ☐ a) seq:2001, payload:100B
- ☐ b) seq:2001, flags:ACK, ack:6101
- ☐ c) seq:2101, flags:ACK, ack:6201
- ☐ d) seq:2101, flags:ACK, ack:6101
- > **15** Segmento enviado por A en el tick 6:
- ☐ a) seq:6201, payload:200B
- ☐ b) seq:6401, payload:100B
- ☐ c) seq:6401, payload:200B
- ☐ d) seq:6401, flags:ACK, ack:2101
- > **16** Segmento enviado por B en el tick 8:
- ☐ a) seq:2001, flags: ack:ACK, ack:6401
- ☐ b) seq:2001, payload:100B
- ☐ c) seq:2101, flags:ACK, ack:6801
- ☐ d) No se envía nada.
- > **17** Segmento enviado por A en el tick 12:
- ☐ a) seq:7001, flags:ACK
- ☐ b) seq:7001, flags:ACK,FIN, ack:2101
- ☐ c) seq:7002, flags:ACK,FIN, ack:2101
- ☐ d) seq:7002, flags:ACK, ack:2102

- 18** [1p] ¿Por qué cuando la carga supera la capacidad de la red, el retardo tiende a infinito?
- ☐ a) Los paquetes tardan más tiempo en alcanzar el destino debido a bucles en las tablas de enrutamiento.
 - ☐ b) Aunque los paquetes no se descarten en ningún router, el tiempo de procesamiento de paquetes en los routers aumenta indefinidamente.
 - ☐ c) El tiempo de propagación de los paquetes a través de los enlaces entre routers tiende a infinito.
 - ☐ d) Los paquetes se descartan en algún router entre el origen y el destino.
- 19** [1p] ¿Con cuál de las siguientes medidas está más relacionada la carga de una red determinada?
- ☐ a) La cantidad total de conexiones TCP establecidas.
 - ☐ b) La cantidad total de hosts activos en cada momento.
 - ☐ c) El porcentaje de ocupación de las colas de los routers.
 - ☐ d) La cantidad de paquetes por segundo reenviados por los routers.
- 20** [1p] ¿Qué indica el campo *window* en la cabecera TCP?
- ☐ a) Longitud del payload (en bytes) de ese segmento.
 - ☐ b) Longitud del payload del segmento recibido (en bytes).
 - ☐ c) Cantidad de datos que el receptor está dispuesto a recibir.
 - ☐ d) Cantidad de datos que el emisor está dispuesto a enviar.