 <b>Universidad de Alcalá</b>	Departamento de Automática Área de Ingeniería Telemática	<h2 style="margin: 0;">Prueba Parcial 3</h2>
APELLIDOS: _____		
NOMBRE: _____ DNI: _____		
ASIGNATURA: <b>ARQUITECTURA DE REDES</b>		TITULACIÓN: <b>GRADO EN INGENIERÍA INFORMÁTICA</b>
FECHA: 12/01//2011	<b>TEST</b>	DURACIÓN: 15 min. (SIN LIBROS)
PUNTUACIÓN: 6/10 Ptos		

NOTA: Cada respuesta correcta suma 0,3 puntos y cada respuesta incorrecta resta 0,3/3 puntos. Las preguntas no contestadas no afectan a la nota.

## Respuestas:

1. A	2. B	3. B	4. C	5. B	6. A	7. A	8. A	9. D	10. A
11. A	12. C	13. B	14. D	15. B	16.D	17. A	18. A	19. D	20. B

1. Indique los valores por los que se identifica un socket TCP:

- a) **Dirección IP origen, puerto origen, dirección IP destino, puerto destino.**
- b) Dirección IP origen, dirección IP destino.
- c) Puerto origen, puerto destino.
- d) Dirección IP destino, puerto destino.

2. Indique cuales de los siguientes campos pertenecen a una cabecera UDP:

- a) Puerto origen, puerto destino, flag SYN, CRC.
- b) **Puerto origen, puerto destino, longitud, CRC.**
- c) Puerto origen, puerto destino, protocolo aplicación, CRC.
- d) Ninguna de las respuestas anteriores es correcta.

3 Sobre los protocolos de bit alternante:

- a) Reciben su nombre del hecho de emplear un bit aleatorio como número de secuencia.
- b) **Emplean checksum, reconocimientos, números de secuencia y temporizadores.**
- c) Emplean checksum, reconocimientos y temporizadores, pero no números de secuencia.
- d) Emplean checksum, reconocimientos y números de secuencia, pero no temporizadores.

4. El protocolo de parada y espera es más ineficiente conforme:

- a) Aumentan las ventanas de emisor y receptor.
- b) Disminuyen las ventanas de emisor y receptor.
- c) **Aumenta el RTT.**
- d) Disminuye el RTT.

5. Sobre el valor del temporizador de retransmisión de TCP:

- a) Debe ser inferior al RTT para que funcione correctamente TCP.
- b) **Emplea el valor de RTT medido para darle valor, dando mayor peso a las muestras recientes que a las antiguas.**
- c) Emplea el valor de RTT medido para darle valor, dando mayor peso a las muestras antiguas que a las recientes.
- d) No considera la desviación de los retardos medidos.

6. Sobre el algoritmo de control de congestión en TCP Vegas:

- a) **Actúa cuando estima la pérdida inminente de segmentos a partir de observar los valores de RTT.**
- b) Actúa a posteriori, una vez que se detecta un evento de pérdida, o presunta pérdida de segmento, independientemente del valor RTT.
- c) Actúa a posteriori, una vez que se detecta un evento de pérdida, o presunta pérdida de segmento, cuando se detecta que el RTT es muy bajo.
- d) Ninguna de las anteriores es correcta.

7. En el modo de recuperación rápida en TCP:

- a) **A la recepción del ACK del segmento que falta, pasa al modo "evitación de la congestión".**
- b) A la recepción del ACK del segmento que falta, pasa al modo "arranque lento".
- c) A la recepción de un nuevo ACK pasa al modo "arranque lento".
- d) A la recepción de un nuevo ACK permanece en el modo "recuperación rápida".

8. Control de Congestión TCP, modo "arranque lento":

- a) **A la recepción de tres ACK's duplicados se pasa al modo "recuperación rápida", si éste está habilitado.**
- b) A la recepción de tres ACK's duplicados se pasa al modo "evitación de la congestión", si éste está habilitado.
- c) A la recepción de tres ACK's duplicados se pasa al modo "recuperación rápida", aunque no esté habilitado.
- d) Ninguna de las anteriores es correcta.

9. En la fase de establecimiento de conexión TCP:

- a) El segmento SYN inicial puede contener datos.
- b) El segmento SYNACK puede contener datos.
- c) El segmento FIN puede contener datos.
- d) **Ninguna de las anteriores es correcta.**

10. El control de flujo en TCP:

- a) **Procedimiento de adaptación de velocidades entre emisor y receptor TCP, que conlleva al control de la velocidad de emisión de segmentos en base a una variable que caracteriza al receptor TCP.**
- b) Procedimiento de adaptación de velocidades entre emisor y receptor TCP, que conlleva al control de la velocidad de emisión de segmentos en base a una variable que caracteriza al emisor TCP.
- c) No es un proceso habitual en el protocolo TCP.
- d) Ninguna de las anteriores es correcta.

11. Temporizador de retransmisión TCP (timeout):

- a) **Su valor se calcula haciendo una estimación del tiempo medio de los RTT, y la desviación de las muestras RTT a partir de este valor medio, dando mayor peso a los valores más recientes de las muestras RTT.**
- b) Su valor se calcula haciendo una estimación del tiempo máximo de los RTT, y la desviación de las muestras RTT a partir de este valor medio, dando mayor peso a los valores más recientes de las muestras RTT.
- c) Su valor se calcula haciendo una estimación del tiempo medio de los RTT, y la desviación de las muestras RTT a partir de este valor medio, dando mayor peso a los valores más antiguos de las muestras RTT.
- d) Ninguna de las anteriores es correcta.


12. Empleando un protocolo de parada y espera, suponga que el host A envía dos segmentos al host B. El host B envía ACKs para cada segmento, el primer ACK se pierde pero el segundo llega al emisor antes de que expire el temporizador del primer paquete.

- a) El host A no retransmitirá ningún paquete.
- b) El host A retransmitirá el primer paquete.
- c) **No puede darse la situación descrita.**
- d) El host A retransmitirá el segundo paquete.

13. Empleando un protocolo de repetición selectiva, suponga que el host A envía dos segmentos al host B. El host B envía ACKs para cada segmento, el primer ACK se pierde pero el segundo llega al emisor antes de que expire el temporizador del primer paquete.
- No puede darse la situación descrita.
  - El host A retransmitirá el primer paquete.**
  - El host A retransmitirá ambos paquetes.
  - El host A retransmitirá el segundo paquete.
14. Un cliente TCP, para establecer una conexión envía un segmento:
- Con el flag SYN activo, N° secuencia inicial y primeros datos.
  - Con los flags SYN y ACK activos, N° secuencia inicial y sin datos.
  - Con el flag ACK activo, N° secuencia inicial y sin datos.
  - Con el flag SYN activo, N° secuencia inicial y sin datos.**
15. Una entidad TCP ha recibido un segmento correctamente pero no tiene datos que enviar al emisor. ¿Qué hace en ese caso?
- No envía un ACK hasta que no tenga datos que enviar.
  - Espera hasta 500 ms a tener datos antes de contestar con un ACK. Si en este tiempo no los tiene y no le llega otro segmento correcto envía el ACK.**
  - La entidad se queda bloqueada, hay que resetearla enviando un segmento con el flag RST activado.
  - Espera un tiempo igual a 2 veces el RTT estimado, y contesta con un ACK.
16. El límite de la velocidad máxima de envío en una conexión TCP:
- Lo fija el receptor con la ventana de recepción, que es función de la capacidad del receptor.
  - Lo fija con la ventana de congestión, que es función de la capacidad ruta entre ambos.
  - Lo fija la ventana de recepción y de congestión, con el máximo de ambas.
  - Lo fija la ventana de recepción y de congestión, con el mínimo de ambas.**
17. TCP ajusta su velocidad de emisión haciendo lo siguiente:
- La incrementa cuando llega un ACK que indica que no hay congestión y la decrementa cuando se pierde un segmento.**
  - La mantiene cuando llega un ACK que indica que no hay congestión y la decrementa cuando se pierde un segmento.
  - La mantiene cuando llega un ACK que indica que no hay congestión y la mantiene cuando se pierde dos segmentos.
  - Ninguna de las anteriores es correcta.
18. El control de congestión en TCP:
- Busca un compromiso entre producir congestión por transmitir demasiado rápido e infrautilizar recursos por ser demasiado cauteloso.**
  - Se inicia al recibir tres ACKs duplicados.
  - El modo de arranque lento es opcional.
  - El modo de evitación de la congestión es opcional.
19. Sobre el formato de cabecera de TCP:
- El campo longitud de cabecera especifica el tamaño de la cabecera medida en octetos.
  - El campo longitud de cabecera especifica el tamaño de la cabecera medida en palabras de 16 bits.
  - La cabecera TCP tiene un tamaño fijo.
  - Se emplean dos octetos para especificar el puerto destino.**

20. Sobre la comparación entre TCP y UDP:

- a) TCP y UDP son protocolos orientados a la conexión.
- b) **TCP y UDP incorporan detección de errores, aunque su uso en UDP es opcional.**
- c) UDP no retransmite los segmentos perdidos, pero sí los erróneos.
- d) UDP, al igual que TCP, realiza un control de flujo del tipo parada y espera.

 <b>Universidad de Alcalá</b>	Departamento de Automática Área de Ingeniería Telemática	<b>Prueba Parcial 3</b>
APELLIDOS: _____		
NOMBRE: _____		DNI: _____
ASIGNATURA: ARQUITECTURA DE REDES		TITULACIÓN: GRADO EN INGENIERÍA INFORMÁTICA
FECHA: 12/01//2011	TEST	DURACIÓN: 15 min. (SIN LIBROS)
PUNTUACIÓN: 6/10 Ptos		

NOTA: Cada respuesta correcta suma 0,3 puntos y cada respuesta incorrecta resta 0,3/3 puntos. Las preguntas no contestadas no afectan a la nota.

## Respuestas:

1. D	2. C	3. A	4. C	5. C	6. C	7. A	8. A	9. D	10. B
11. C	12. B	13. B	14. B	15. B	16. D	17. D	18. D	19. A	20. C

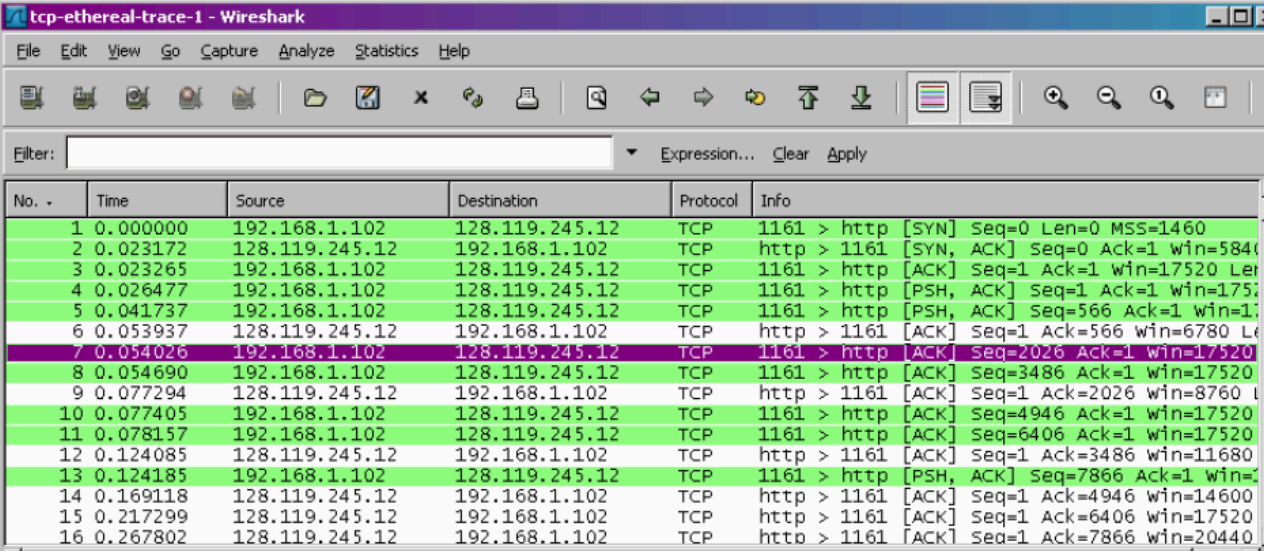
1. Indique los valores por los que se identifica un socket UDP:

- a) Dirección IP origen, puerto origen, dirección IP destino, puerto destino.
- b) Dirección IP origen, dirección IP destino.
- c) Puerto origen, puerto destino.
- d) Dirección IP destino, puerto destino.

2. Entre los protocolos de aplicación que utilizan UDP podemos señalar:

- a) HTTP, FTP, NTP.
- b) SNMP, SMTP, FTP.
- c) NTP, TFTP, DNS.
- d) FTP, HTTP, MIME, TFTP.

3. La siguiente figura muestra una captura de tramas utilizando Wireshark. Sobre las tres primeras tramas podemos afirmar:



No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	1161 > http [SYN] Seq=0 Len=0 MSS=1460
2	0.023172	128.119.245.12	192.168.1.102	TCP	http > 1161 [SYN, ACK] Seq=0 Ack=1 win=5840
3	0.023265	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=1 Ack=1 win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	1161 > http [PSH, ACK] Seq=1 Ack=1 win=17520
5	0.041737	192.168.1.102	128.119.245.12	TCP	1161 > http [PSH, ACK] Seq=566 Ack=1 win=17520
6	0.053937	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=566 win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=2026 Ack=1 win=17520
8	0.054690	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=3486 Ack=1 win=17520
9	0.077294	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=2026 win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=4946 Ack=1 win=17520
11	0.078157	192.168.1.102	128.119.245.12	TCP	1161 > http [ACK] Seq=6406 Ack=1 win=17520
12	0.124085	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=3486 win=11680
13	0.124185	192.168.1.102	128.119.245.12	TCP	1161 > http [PSH, ACK] Seq=7866 Ack=1 win=17520
14	0.169118	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=4946 win=14600
15	0.217299	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=6406 win=17520
16	0.267802	128.119.245.12	192.168.1.102	TCP	http > 1161 [ACK] Seq=1 Ack=7866 win=20440

- a) Se trata de un establecimiento de conexión TCP.
- b) Se trata de un cierre de conexión TCP.
- c) Se trata de un intercambio de tramas entre los puertos UDP.
- d) Ninguna de las respuestas anteriores es correcta.

4. Los protocolos de transferencia de datos, para recuperarse de la pérdida de paquetes emplean:
- Checksum (también llamado CRC).
  - Reconocimientos positivos y/o negativos.
  - Temporizadores (timeouts).**
  - Ninguna de las anteriores es correcta.
5. En el arranque lento ¿Cuál es el valor por el que se modifica la ventana de transmisión?
- Se le añade el valor de la ventana inicial cada vez.
  - Se incrementa 100 octetos cada RTT.
  - Se incrementa 1 MSS cada primer ACK de un segmento transmitido.**
  - Se incrementa 100 bits cada RTT.
6. En el cálculo del valor de *timeout* asociado a la emisión de un segmento se tiene en cuenta:
- El valor del último RTT calculado para el último segmento transmitido pero no el promedio de RTT anteriores.
  - El valor promedio de valores RTT anteriores sin incluir el último RTT medido.
  - El valor del último RTT calculado para el último segmento transmitido y el promedio de RTT anteriores.**
  - Ninguna de las anteriores es correcta.
7. Los procedimientos de control de congestión en las versiones TCP Tahoe y Reno se caracterizan por:
- Actuar a posteriori, una vez detectada la pérdida o presunta pérdida de un segmento.**
  - Actuar con antelación a la ocurrencia de un evento de pérdida o presunta pérdida de segmento.
  - Actuar con antelación o a posteriori a la ocurrencia de un evento de pérdida o presunta pérdida de segmento, siempre y cuando se esté en el modo Arranque Lento.
  - Ninguna de las anteriores es correcta.
8. En el modo "recuperación rápida" de TCP:
- A la recepción de un ACK duplicado la variable ventana de congestión (VC) se incrementa un MSS**
  - A la recepción de un ACK duplicado la variable ventana de congestión (VC) se incrementa una fracción de MSS.
  - A la recepción de un ACK duplicado la variable ventana de recepción (VR) se incrementa un MSS.
  - A la recepción de un ACK duplicado la variable ventana de recepción (VR) se incrementa una fracción de MSS.
9. Control de Congestión TCP, modo "evitación de la congestión":
- A la recepción de tres ACK's duplicados se pasa al modo "recuperación rápida" si se emplea TCP Tahoe.
  - A la recepción de dos ACK's duplicados se pasa al modo "recuperación rápida", si éste está habilitado.
  - A la recepción de tres ACK's duplicados permanece en el modo "evitación de la congestión".
  - Ninguna de las anteriores es correcta.**
10. Control de Congestión TCP:
- Quando vence el temporizador de retransmisión se pasa al modo "arranque lento", y la variable VC se disminuye a la mitad.
  - Quando vence el temporizador de retransmisión se pasa al modo "arranque lento", y la variable VC se hace igual a 1 MSS.**
  - Quando vence el temporizador de retransmisión se pasa al modo "evitación de la congestión", y la variable VC se hace igual a 1 MSS.
  - Ninguna de las anteriores es correcta.

11. Control de flujo TCP:

- a) Procedimiento mediante el cual receptor notifica a emisor el estado de su buffer de recepción, por intermedio de la variable "ventana del emisor".
- b) Procedimiento mediante el cual receptor notifica a emisor el estado de su buffer de recepción, por intermedio de la variable "umbral".
- c) **Procedimiento mediante el cual receptor notifica a emisor el estado de su buffer de recepción, por medio de la variable "ventana de recepción".**
- d) Ninguna de las anteriores es correcta.

12. En relación con el cierre de una conexión TCP:

- a) Un cierre ordenado de la conexión en ambos sentidos sólo requiere el intercambio de dos segmentos.
- b) **El cliente responde con un segmento con el bit ACK activo a los segmentos con el bit FIN activo recibidos.**
- c) El cliente cierra la conexión inmediatamente después de enviar un segmento con el bit ACK activo.
- d) Ninguno de las anteriores es correcta.

13. El algoritmo de retransmisión rápida:

- a) Permite identificar la pérdida de un segmento y retransmitirlo, pero no antes de que venza el timeout.
- b) **Se activa con la llegada al emisor de tres segmentos seguidos con el mismo valor de ACK, cuando este ha seguido enviando segmentos de datos.**
- c) Permite evitar la pérdida de un segmento y retransmitirlo, pero no antes de que venza el timeout.
- d) Ninguna de las anteriores es correcta.

14. ¿En el mecanismo de control de congestión TCP?

- a) La red IP subyacente siempre informa a los terminales de la existencia de congestión de red.
- b) **La red IP no informa de congestión por lo que los terminales deben deducir cuando hay congestión.**
- c) La velocidad de transmisión la fijará exclusivamente el receptor que es quien puede tener limitaciones en la capacidad de recepción.
- d) Ninguna de las anteriores es correcta.

15. Cuando hay congestión en un router y se descarta un datagrama:

- a) El receptor manda un mensaje especial al emisor para que baje la velocidad.
- b) **El emisor recibirá tres ACK duplicados o le vencerá un temporizador y bajará la velocidad.**
- c) El emisor entenderá que hay congestión sólo cuando venza el temporizador asociado al datagrama perdido.
- d) El receptor entiende que hay congestión y reduce la *ventana del receptor* del emisor.

16. El arranque lento TCP termina:

- a) Sólo termina cuando vence el timeout.
- b) Sólo termina cuando se reciben tres ACK duplicados.
- c) Sólo termina cuando la ventana de congestión alcanza el valor del umbral.
- d) **Cuando se da cualquiera de las tres causas anteriores.**

17. TCP es un protocolo:

- a) De tipo Retroceder N (GBN).
- b) De tipo Repetición Selectiva (SR).
- c) De tipo Parada y Espera (S&W).
- d) **Ninguna de las anteriores es correcta.**

18. En cuanto a la comparación entre TCP y UDP:

- a) Tanto TCP como UDP realizan control de flujo, siendo en este último caso mediante el uso de "parada y espera".
- b) A pesar de que TCP es más complejo que UDP, ambos tienen un tamaño de cabecera de 8 bytes.
- c) UDP no realiza segmentación, pero sí reensamblado.
- d) **UDP no es un protocolo orientado a conexión.**

19. Una entidad TCP retransmite un segmento ( $N^{\circ}$  seq = 95; tamaño = 5 octetos) al haber expirado el timeout y mientras los transmite recibe un segmento con ACK = 100.

- a) **Se trata de un caso de timeout prematuro. Volverá a recibir un segmento con ACK = 100.**
- b) Debería haber recibido un segmento con ACK = 101.
- c) No debe haber recibido ningún ACK hasta que el segmento llegue al receptor y éste conteste.
- d) Esto indica un bloqueo del protocolo.

20. Sobre el protocolo UDP:

- a) La carga útil puede tener un tamaño de hasta  $(2^{16} - 1)$  bytes.
- b) El número de puerto máximo es  $2^{16} - 1$ , que debe repartirse entre UDP y TCP.
- c) **El puerto origen en UDP se emplea para una posible respuesta.**
- d) Ninguna de las anteriores es correcta.



APELLIDOS: _____			
NOMBRE: _____		DNI: _____	
ASIGNATURA: ARQUITECTURA DE REDES		TITULACIÓN: GRADO EN INGENIERÍA INFORMÁTICA	
FECHA: 12/01/2011	PROBLEMA	DURACIÓN: 50 min. (SIN LIBROS)	PUNTUACIÓN: 4/10 Ptos

### EJERCICIO 1 (2 puntos)

Desde un PC se desean enviar 8 segmentos de la capa de transporte hasta un servidor utilizando una técnica de rechazo simple con 3 bits para la numeración de paquetes y tamaño máximo de ventana de transmisión.

Los tiempos de recepción de paquetes de datos en la capa de transporte son iguales a 20 ms, el tiempo de transmisión de segmentos de datos, medido en la capa de transporte es igual a 10 ms y el tiempo de transmisión de las confirmaciones (positivas) se considera despreciable y el temporizador de retransmisión tiene un valor de 50 ms.

Suponga que la transferencia de datos se produce del siguiente modo:

- ☐ El segundo paquete de datos (original) que se envía se recibe con error y su retransmisión se recibe correctamente.
- ☐ Se pierde el cuarto paquete de datos (original) que se envía y su retransmisión se recibe correctamente.
- ☐ Se pierde la confirmación del último paquete de datos (original).

Complete el siguiente diagrama de tiempos en el que se muestra el intercambio de segmentos de datos y confirmaciones que se produce en la capa de transporte, indicando el tiempo total empleado para completar el intercambio.

### EJERCICIO 2 (2 puntos)

Suponga una máquina A, cuya entidad TCP establece una conexión con la entidad TCP de otra máquina B. Ambas entidades, A y B, manejan tamaños máximos de segmento a enviar (MSS) de 100 octetos. Tras el establecimiento de conexión, A enviará a B, 700 octetos de datos TCP. Tras el envío de datos A finaliza la conexión con B.

Suponiendo que:

- El envío de cada segmento se produce en el inicio de un 'tic' de reloj.
- Cada segmento tarda medio 'tic' en llegar a la entidad contraria.
- B no tiene información que enviar y se limita a confirmar la recepción de segmentos de A de acuerdo con la siguiente política:
  - o De forma inmediata durante la fase de establecimiento y cierra de conexión.
  - o Se recibe segmento con N° de secuencia esperado (anteriores ya reconocidos): ACK retardado. Se espera 1 'tic' la llegada de otro segmento en secuencia. Si no llega, se envía ACK.
  - o Se recibe segmento con N° de secuencia esperado y hay otro segmento en orden esperando transmisión de un ACK: ACK único acumulativo. Se reconocen ambos segmentos ordenados.
  - o Se recibe segmento fuera de secuencia, con N° mayor que el esperado. Se detecta un "hueco": ACK duplicado. Se envía de inmediato ACK con N° de secuencia del siguiente octeto esperado (límite inferior del "hueco")
  - o Se recibe segmento que completa parcial o totalmente "hueco" en los datos recibidos: ACK inmediato. Se envía de inmediato ACK si el segmento comienza en el límite inferior del "hueco".
- Tras cada confirmación B permite expandir la ventana de A a su tamaño original.
- A comienza con un SN=999 y B con SN=1999.
- Al acabar de transmitir, A inicia el cierre de la conexión.
- Considere que tanto la entidad TCP A como la B manejan ventanas de recepción de 800 octetos y se emplea una técnica de control de congestión de arranque lento.

Dibuje un diagrama de envío y recepción en el que muestre, para cada segmento, los campos más significativos que procedan: SN(nro de secuencia), AN(nro de reconocimiento), W(ventana permitida para la entidad contraria), F=flags activos, MSS(máximo tamaño segmento) y Datos (tamaño, en octetos, del campo de datos enviados).

APELLIDOS: \_\_\_\_\_  
NOMBRE: \_\_\_\_\_ DNI: \_\_\_\_\_

ASIGNATURA: ARQUITECTURA DE REDES

TITULACIÓN: GRADO EN INGENIERÍA INFORMÁTICA

FECHA: 12/01/2011

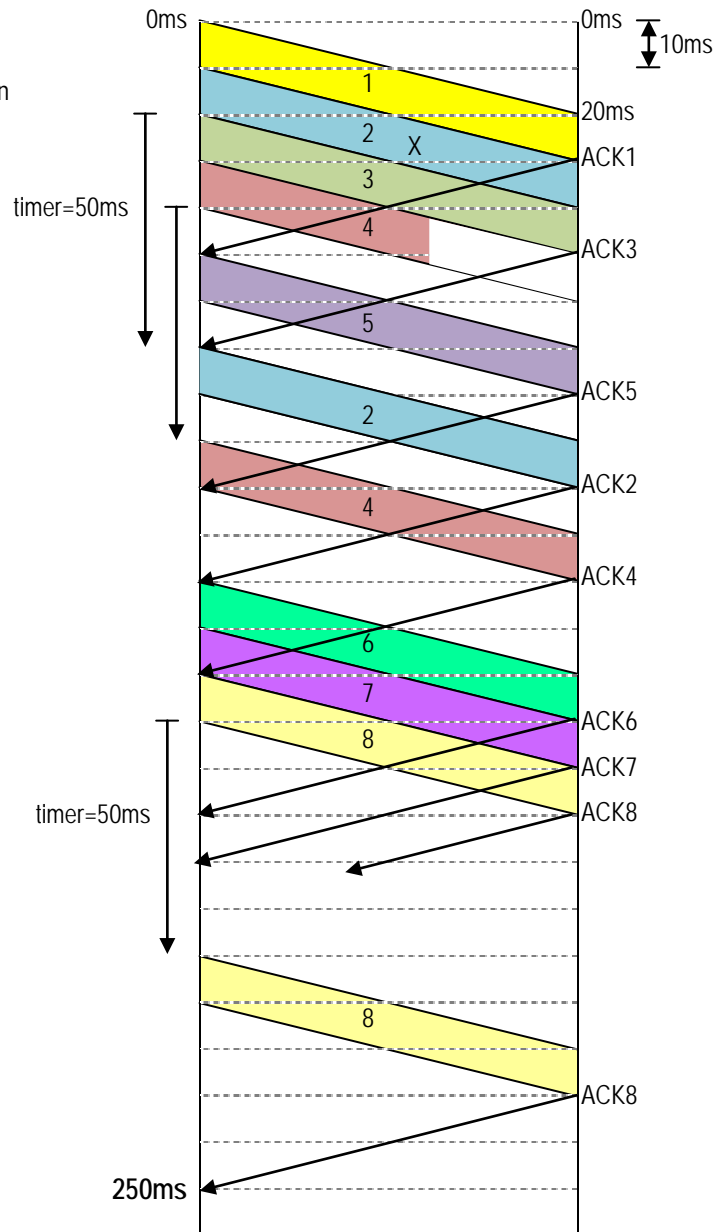
PROBLEMA

DURACIÓN: 50 min. (SIN LIBROS)

PUNTUACIÓN: 4/10 Ptos

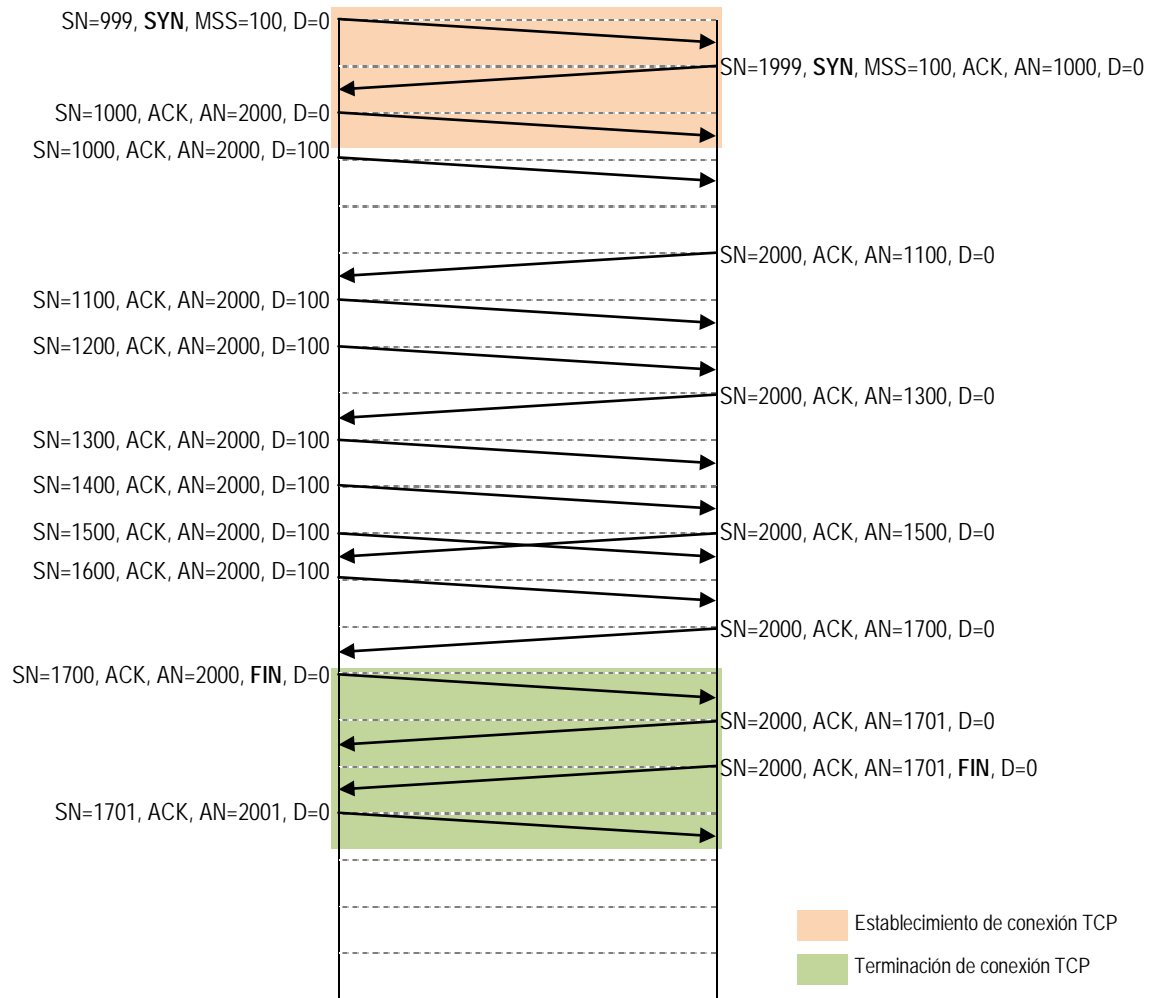
## EJERCICIO 1:

Rechazo simple (o repetición selectiva):  $W_{\max} = 2^{n-1} = 2^{3-1} = 4$



## EJERCICIO 2:

NOTA: En todos los casos se anuncia W=800, por lo que no se indica por brevedad en la respuesta



APELLIDOS: _____			
NOMBRE: _____		DNI: _____	
ASIGNATURA: ARQUITECTURA DE REDES		TITULACIÓN: GRADO EN INGENIERÍA INFORMÁTICA	
FECHA: 12/01/2011	PROBLEMA	DURACIÓN: 50 min. (SIN LIBROS)	PUNTUACIÓN: 4/10 Ptos

### EJERCICIO 1 (2 puntos)

Desde un PC se desean enviar 8 segmentos de la capa de transporte hasta un servidor utilizando una técnica de rechazo "vuelta atrás N" con 3 bits para la numeración de paquetes y tamaño máximo de ventana de transmisión.

Los tiempos de recepción de paquetes de datos en la capa de transporte son iguales a 20 ms, el tiempo de transmisión de segmentos de datos, medido en la capa de transporte es igual a 10 ms y el tiempo de transmisión de las confirmaciones (positivas) se considera despreciable y el temporizador de retransmisión tiene un valor de 50 ms.

Suponga que la transferencia de datos se produce del siguiente modo:

- ☐ El segundo paquete de datos (original) que se envía se recibe con error y su retransmisión se recibe correctamente.
- ☐ Se pierde el cuarto paquete de datos (original) que se envía y su retransmisión se recibe correctamente.
- ☐ Se pierde la confirmación del último paquete de datos (original).

Complete el siguiente diagrama de tiempos en el que se muestra el intercambio de segmentos de datos y confirmaciones que se produce en la capa de transporte, indicando el tiempo total empleado para completar el intercambio.

### EJERCICIO 2 (2 puntos)

Suponga una máquina A, cuya entidad TCP establece una conexión con la entidad TCP de otra máquina B. Ambas entidades, A y B, manejan tamaños máximos de segmento a enviar (MSS) de 100 octetos. Tras el establecimiento de conexión, A enviará a B, 700 octetos de datos TCP. Tras el envío de datos A finaliza la conexión con B.

Suponiendo que:

- El envío de cada segmento se produce en el inicio de un 'tic' de reloj.
- Cada segmento tarda medio 'tic' en llegar a la entidad contraria.
- B no tiene información que enviar y se limita a confirmar la recepción de segmentos de A de acuerdo con la siguiente política:
  - o De forma inmediata durante la fase de establecimiento y cierra de conexión.
  - o Se recibe segmento con N° de secuencia esperado (anteriores ya reconocidos): ACK retardado. Se espera 1 'tic' la llegada de otro segmento en secuencia. Si no llega, se envía ACK.
  - o Se recibe segmento con N° de secuencia esperado y hay otro segmento en orden esperando transmisión de un ACK: ACK único acumulativo. Se reconocen ambos segmentos ordenados.
  - o Se recibe segmento fuera de secuencia, con N° mayor que el esperado. Se detecta un "hueco": ACK duplicado. Se envía de inmediato ACK con N° de secuencia del siguiente octeto esperado (límite inferior del "hueco")
  - o Se recibe segmento que completa parcial o totalmente "hueco" en los datos recibidos: ACK inmediato. Se envía de inmediato ACK si el segmento comienza en el límite inferior del "hueco".
- Tras cada confirmación B permite expandir la ventana de A a su tamaño original.
- A comienza con un SN=499 y B con SN=999.
- Al acabar de transmitir, A inicia el cierre de la conexión.
- Considere que tanto la entidad TCP A como la B manejan ventanas de recepción de 200 octetos y se emplea una técnica de control de congestión de arranque lento.

Dibuje un diagrama de envío y recepción en el que muestre, para cada segmento, los campos más significativos que procedan: SN(nro de secuencia), AN(nro de reconocimiento), W(ventana permitida para la entidad contraria), F=flags activos, MSS(máximo tamaño segmento) y Datos (tamaño, en octetos, del campo de datos enviados).

APELLIDOS: \_\_\_\_\_  
NOMBRE: \_\_\_\_\_ DNI: \_\_\_\_\_

ASIGNATURA: **ARQUITECTURA DE REDES**

TITULACIÓN: GRADO EN INGENIERÍA INFORMÁTICA

FECHA: 12/01/2011

## PROBLEMA

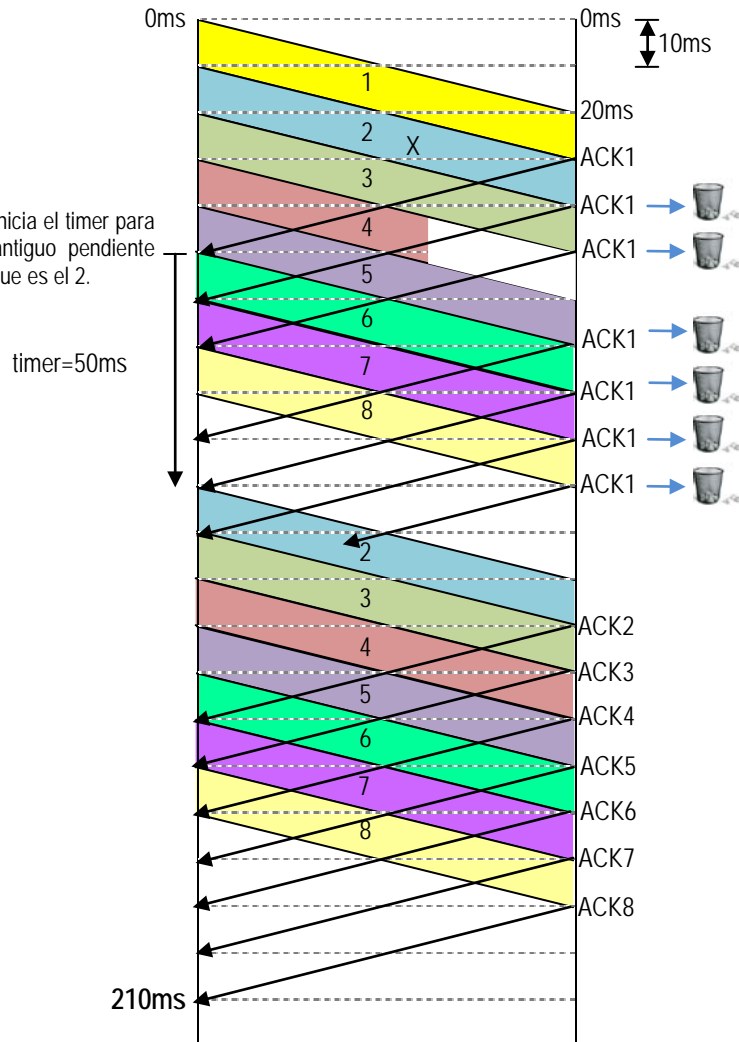
**DURACIÓN: 50 min. (SIN LIBROS)**

PUNTUACIÓN: 4/10 Ptos

## EJERCICIO 1:

GBN:  $W_{\max} = 2^n - 1 = 2^3 - 1 = 7$

En este punto se reinicia el timer para el segmento más antiguo pendiente de reconocimiento, que es el 2.



## EJERCICIO 2:

NOTA: En todos los casos se anuncia W=200, por lo que no se indica por brevedad en la respuesta

