



# HY335b

3<sup>ο</sup> Φροντιστήριο

Μπαμιεδάκης Μιχάλης

# Ερωτήσεις

Ε: Γιατί χρειαζόνταν να εισάγουμε αριθμούς ακολουθίας (sequence numbers) στην επικεφαλίδα του TCP;

Α: Για να γνωρίζουμε αν ένα πακέτο, το οποίο φτάνει στο παραλήπτη, περιέχει καινούργια δεδομένα ή είναι αναμετάδοση. Επίσης, για να μπορούμε να βάλουμε τα δεδομένα που έχουν παραληφθεί σε σωστή σειρά.

Ε: Γιατί χρειάζεται να εισάγουμε timers;

Α: Για να χειριστούμε απώλειες λόγω καναλιού. Αν το ACK για ένα πακέτο που έχει μεταδοθεί, δεν ληφθεί μέσα στο χρονικό όριο που θέτει ο timer, τότε υποθέτουμε πως το πακέτο (ή το ACK) έχει χαθεί. Έτσι κάνουμε αναμετάδοση του πακέτου.

# Ερωτήσεις

Ε: Το TCP περιμένει μέχρι να λάβει 3 duplicate ACKs πριν να επιχειρήσει μια γρήγορη επαναμετάδοση. Γιατί πιστεύετε ότι οι σχεδιαστές του TCP επέλεξαν να μην επιχειρεί το TCP, μια γρήγορη επαναμετάδοση μετά από τη λήψη του πρώτου duplicate ACK;

Απάντηση:

Υποθέστε ότι έρχονται τα πακέτα  $n$ ,  $n+1$  και  $n+2$  και ότι το πακέτο  $n$  φτάνει στο παραλήπτη και στέλνεται το ACK. Αν η σειρά που παραλαμβάνονται τα πακέτα  $n+1$  και  $n+2$  αλλάξει, τότε ο παραλήπτης θα στείλει ένα duplicate ACK για το  $n$  και θα έπρεπε να μεταδώσουμε τα  $n+1$  και  $n+2$  ξανά.

Άρα περιμένοντας για τρία duplicate ACKs, για να μεταδώσουμε ξανά τα πακέτα από το  $n+1$  θα πρέπει να φτάσουν στο παραλήπτη 2 πακέτα μετά το πακέτο  $n$  ΚΑΙ να μην έχει φτάσει το πακέτο  $n+1$ .

# Πρόβλημα 1

Υποθέστε ότι μεταφέρουμε ένα τεράστιο αρχείο μεγέθους  $L$  bytes, από το τερματικό A στο τερματικό B. Υποθέστε  $MSS$  536 bytes. Θυμηθείτε ότι το πεδίο sequence number του TCP έχει 4 bytes.

➤ Ποια είναι η μέγιστη τιμή του  $L$ , ώστε να μην εξαντληθούν τα sequence numbers του TCP;

Απάντηση:

Ο αριθμός των πιθανών αριθμών ακολουθίας είναι  $2^{32} = 4.294.967.296$ . Ο αριθμός ακολουθίας δεν αυξάνεται κατά ένα για κάθε segment, αλλά αυξάνεται κατά των αριθμό των αποσταλθέντων bytes. Έτσι το μέγιστο μέγεθος αρχείου που μπορεί να σταλεί από τον A στο B είναι 4.29487 Gbytes.

# Πρόβλημα 1

Για το L του προηγούμενου ερωτήματος, βρείτε το χρόνο που χρειάζεται για να μεταδοθεί το αρχείο. Υποθέστε προσθήκη 66 bytes σε κάθε segment από τα επίπεδα μεταφοράς, δικτύου και ζευξης. Το πακέτο στέλνεται μέσω μιας ζεύξης 155 Mbps.

Απάντηση:

Ο αριθμός των segments είναι  $2^{32}/536=8,012,999$ , επίσης προστίθεται 66 bytes επικεφαλίδας σε κάθε segment, άρα 528,857,934 bytes επικεφαλίδας. Ο συνολικός αριθμός μεταδιδόμενων bytes είναι  $4,294,967,232 + 528,857,934 = 4,824 \times 10^9$  bytes. Έτσι θα έπαιρνε 249 δευτερόλεπτα για να μεταδοθεί το αρχείο σε ένα link 155 Mbps.



# Πρόβλημα 2

Τα τερματικά A και B επικοινωνούν μέσω μιας σύνδεσης TCP. Το τερματικό B έχει ήδη λάβει από το A όλα τα bytes έως το byte 126. Υποθέστε ότι έπειτα ο A στέλνει δύο segments το ένα πίσω από το άλλο. Το πρώτο και το δεύτερο segment περιέχουν 70 και 50 bytes αντίστοιχα. Το source port είναι 302 και το destination port είναι 80. Ο B στέλνει acknowledgement όποτε λαμβάνει segment από τον A.

➤ Στο δεύτερο segment από το A στο B, ποιά είναι ο αριθμός sequence, το source port και το destination port.

Απάντηση:

Στο δεύτερο segment από το A στο B το sequence number είναι 197, το source port number είναι 302 και το destination port number είναι 80.

# Πρόβλημα 2

➤ Αν το πρώτο segment φτάσει στο παραλήπτη πριν το δεύτερο segment, στο acknowledgement του πρώτου segment ποιος θα είναι ο αριθμός acknowledgement, το source port και destination port.

Απάντηση:

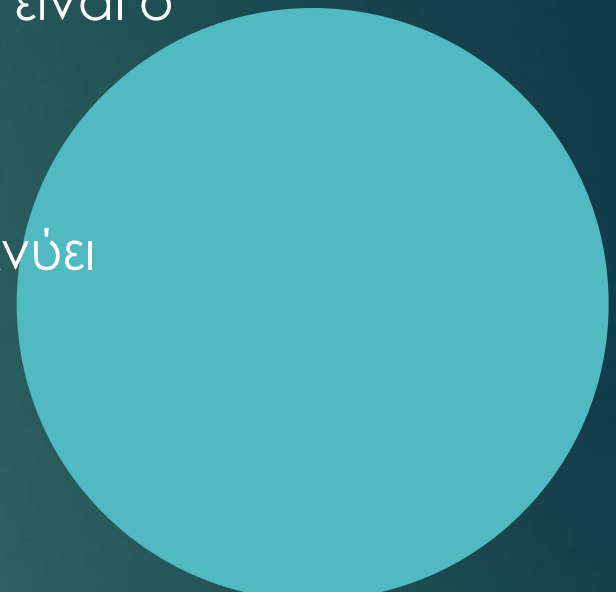
Ο αριθμός acknowledgement θα είναι 197, το source port θα είναι το 80 και το destination port θα είναι το 302.

# Πρόβλημα 2

► Εάν το δεύτερο segment, φτάσει στο δέκτη πριν το πρώτο, στο acknowledgement του πρώτου segment που έχει φτάσει, ποιός είναι ο αριθμός του acknowledgement.

Απάντηση:

Ο αριθμός acknowledgement είναι 127, πράγμα που υποδεικνύει πως το B περιμένει τα bytes από 127 και μετά.





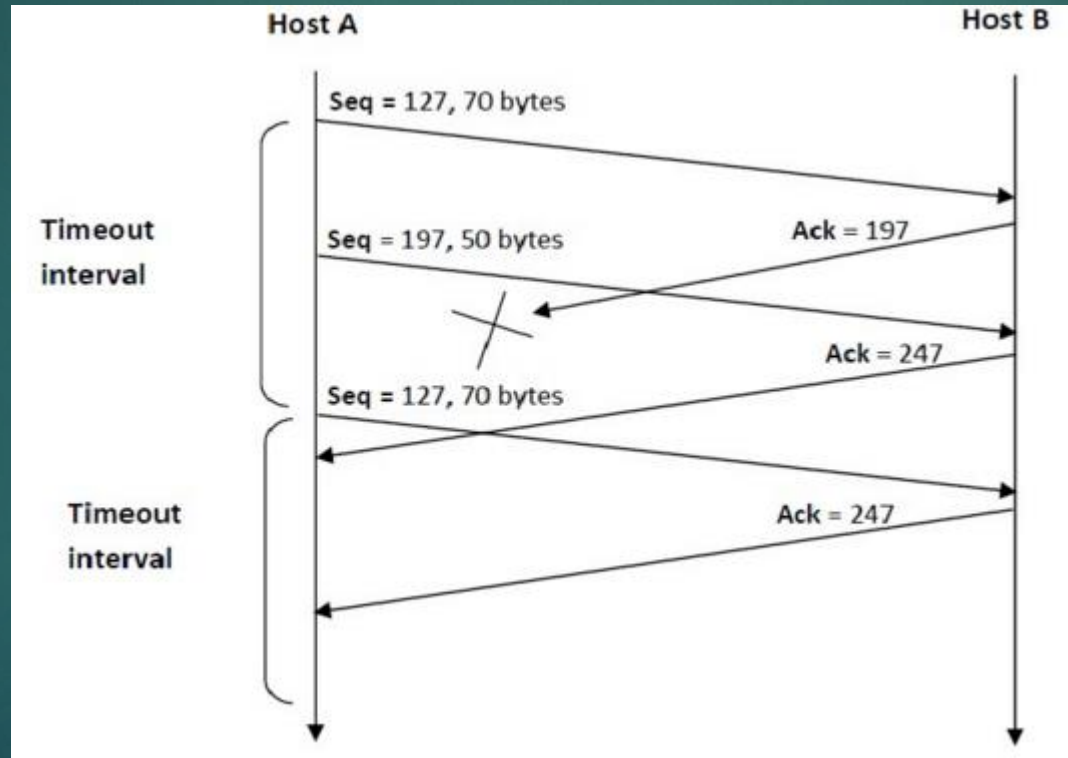
# Πρόβλημα 2

► Υποθέστε ότι δύο segments στέλνονται απο το A και φτάνουν σε σωστή σειρά στο B. Το πρώτο acknowledgement φτάνει, μετά το πρώτο time-out. Σχεδιάστε ένα χρονικό διάγραμμα, το οποίο δείχνει όλα τα segments που στέλνονται, καθώς και τα acknowledgements. Για κάθε segment στο σχήμα σας, δείξτε τον αριθμό sequence και τον αριθμό των bytes δεδομένων. Για κάθε acknowledgement που εισάγετε, δείξτε τον αριθμό acknowledgement.



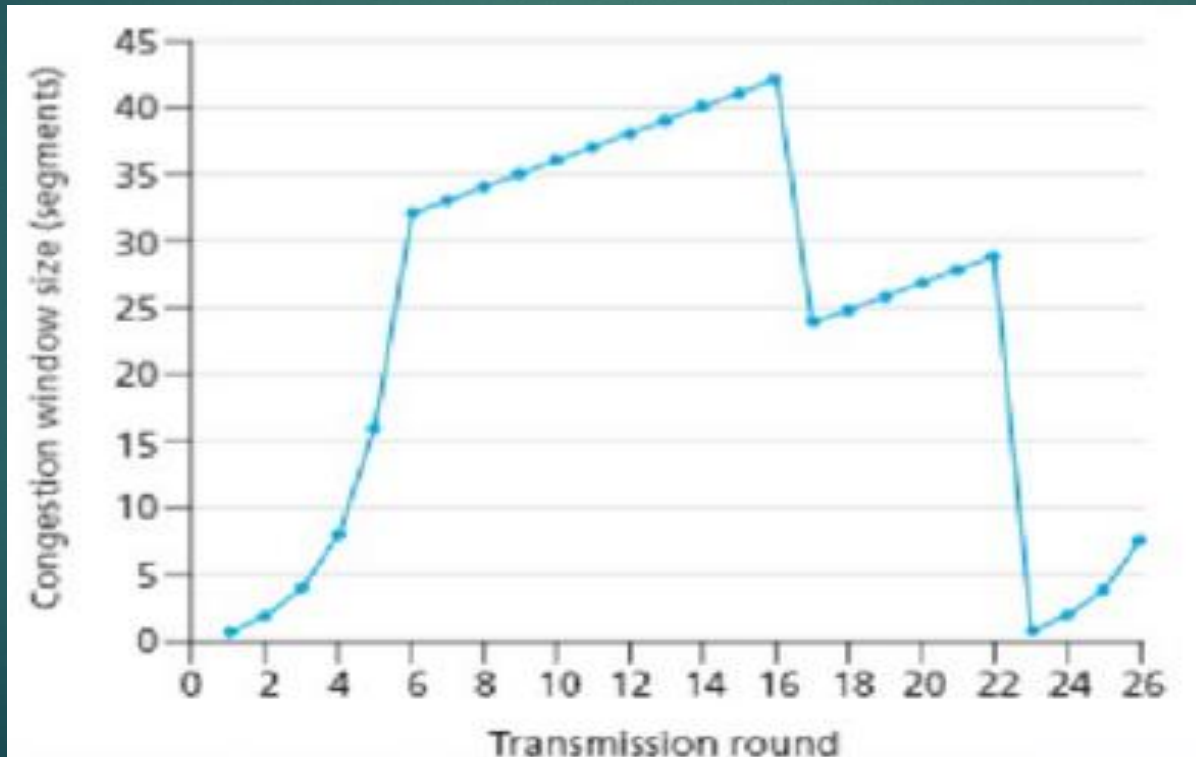
# Πρόβλημα 2

Απάντηση:

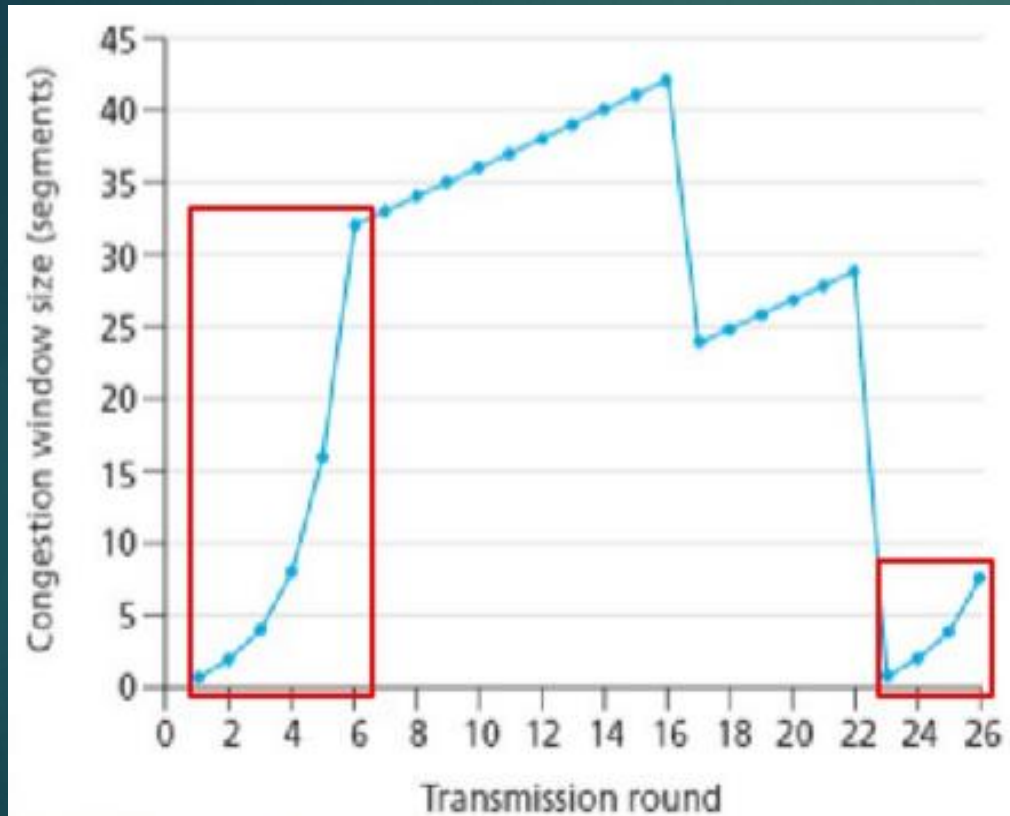


# Πρόβλημα 3

Υποθέστε ότι το παρακάτω διάγραμμα δείχνει τη συμπεριφορά του TCP για ένα τερματικό. Απαντήστε τις παρακάτω ερωτήσεις. Σε όλες τις περιπτώσεις αιτιολογήστε την απάντησή σας.



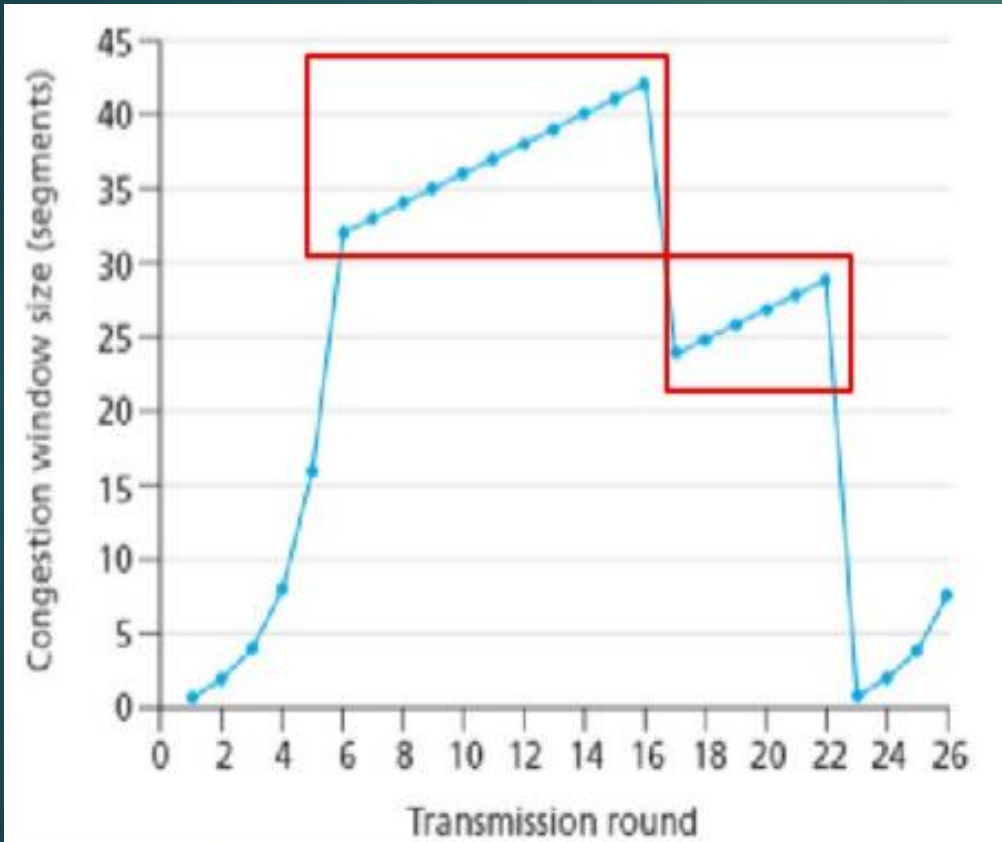
# Πρόβλημα 3



➤ Δείξτε τα χρονικά διαστήματα όπου το συμβαίνει η αργή εκκίνηση του TCP.

Απάντηση:  
Η αργή εκκίνηση του TCP εμφανίζεται στα χρονικά διαστήματα [1,6] και [23,26]

# Πρόβλημα 3



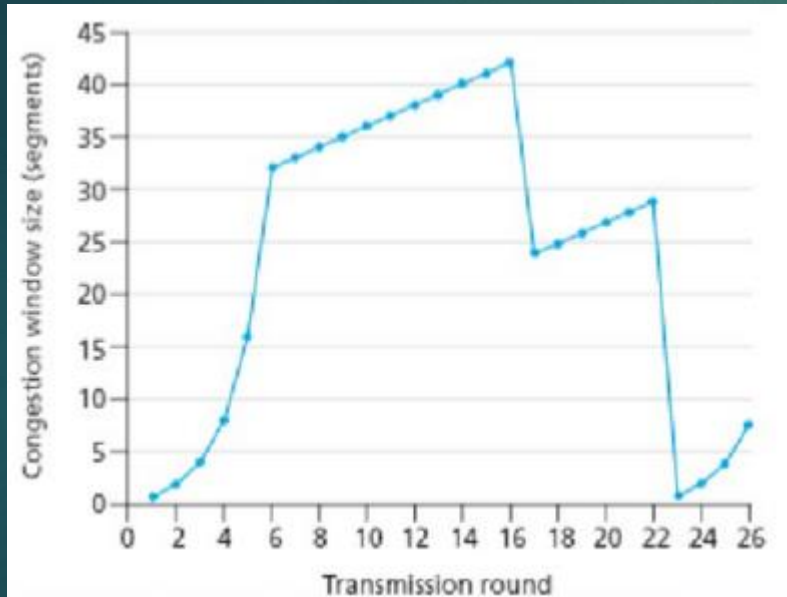
➤ Αναγνωρίστε τα χρονικά διαστήματα όπου το TCP κάνει αποφυγή συμφόρησης. Μετά το 16<sup>ο</sup> γύρο μετάδοσης, πως αναγνωρίζεται η απώλεια πακέτων;

Απάντηση:

Μετά το 16 γύρο μετάδοσης, η απώλεια πακέτων αναγνωρίζεται από τρία duplicate ACKs.



# Πρόβλημα 3



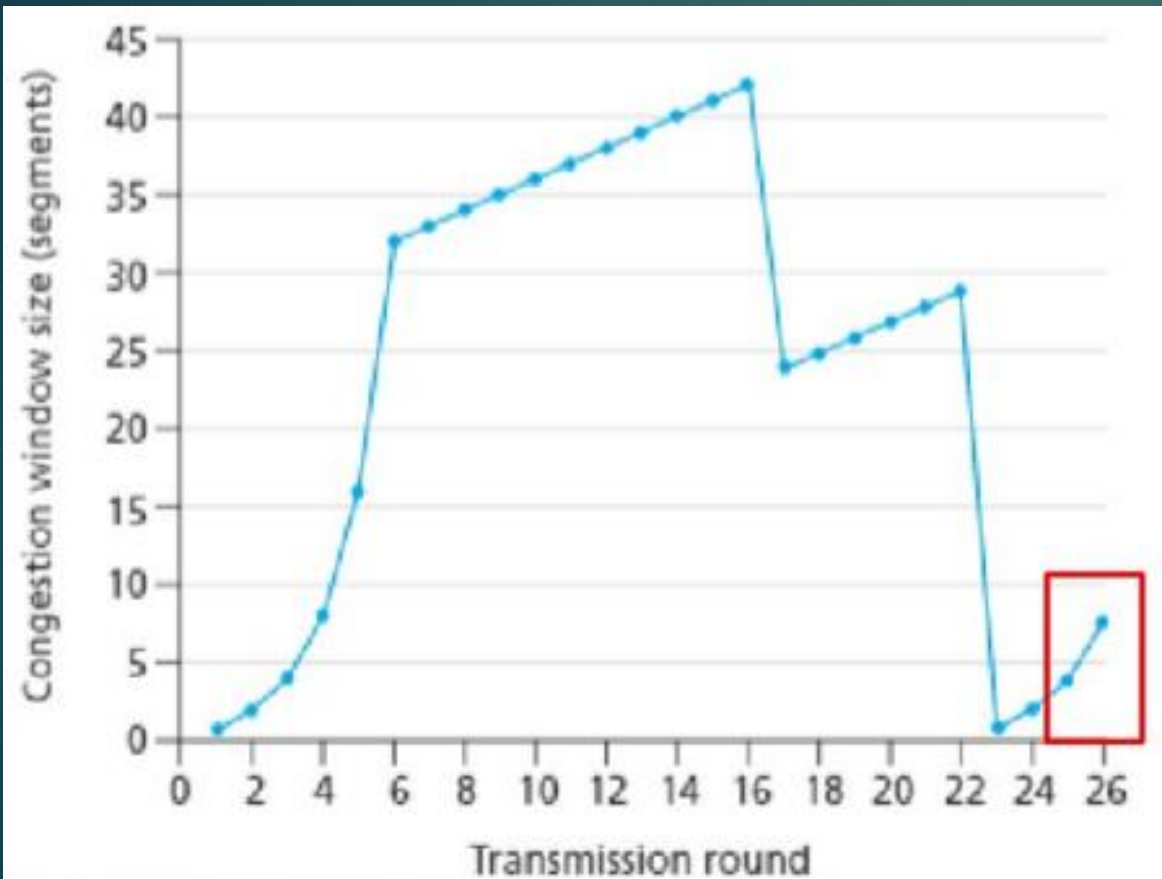
Σε ποιο γύρο μετάδοσης στέλνεται το 70<sup>ο</sup> segment;

Απάντηση:

Κατά το πρώτο γύρο μετάδοσης στέλνεται το πρώτο πακέτο.

Τα πακέτα 2-3 στέλνονται στο δεύτερο γύρο.  
4-7 στο τρίτο, 8-15 στο τέταρτο, 16-31 στο πέμπτο,  
32-63 στον έκτο και 64-96 στον έβδομο. Άρα το 70<sup>ο</sup> Segment στέλνεται στον 7<sup>ο</sup> γύρο μετάδοσης.

# Πρόβλημα 3



Υποθέτοντας ότι μια απώλεια πακέτων εντοπίζεται μετά τον 26<sup>ο</sup> γύρο, μέσω τριπλού duplicate ACK. Ποιό θα είναι το μέγεθος του παραθύρου αποφυγής συμφόρησης.

Απάντηση:  
Το παράθυρο συμφόρησης θα μειωθεί στο 4.

# Πρόβλημα 4

► Θεωρήστε ότι μία TCP σύνδεση χρησιμοποιεί ένα link 10Mbps το οποίο δεν συσσωρεύει (buffer) δεδομένα. Υποθέστε ότι ο αποστολέας TCP έχει ένα πολύ μεγάλο αρχείο να στείλει στον παραλήπτη και ότι ο buffer του παραλήπτη είναι πολύ μεγαλύτερος από το παράθυρο συμφόρησης. Επίσης κάνουμε τις ακόλουθες υποθέσεις:

Κάθε TCP segment έχει μέγεθος 1500 bytes(MSS), η αμφίδρομη καθυστέρηση διάδοσης της σύνδεσης είναι 100 ms. Τέλος, η σύνδεση αυτή είναι πάντα σε φάση αποφυγής συμφόρησης.

# Πρόβλημα 4

➤ Ποιό είναι το μέγιστο μέγεθος παραθύρου (segments) που μπορεί να επιτύχει αυτή η σύνδεση;

Απάντηση:

Έστω ότι  $W$  segments είναι το μέγιστο μέγεθος παραθύρου.

Τότε,  $W \cdot MSS / RTT = 10 \text{ Mbps}$ . Έτσι έχουμε

$$W \cdot 1500 \cdot 8 / 0.1 = 10 \cdot 10^6.$$

$$W = 84 \text{ (ανώφλι του } 83.3 \text{)}.$$



# Πρόβλημα 4

➤ Ποιό είναι το μέσω μέγεθος παραθύρου και το μέσο throughput (σε bps) αυτής της TCP σύνδεσης;

Απάντηση:

Το μέσο μήκος παραθύρου είναι  $0.75 * W = 63$  segments. Άρα το μέσο throughput είναι  $63 * 1500 * 8 / 0.1 = 7.56 \text{ Mbps}$ .



# Πρόβλημα 5

➤ Στο GBN με μέγεθος παραθύρου 3, αν κάθε 5<sup>ο</sup> πακέτο που μεταδίδεται χάνεται, και αν θέλουμε να στείλουμε 10 πακέτα, τότε πόσες μεταδόσεις χρειάζονται, μέχρι ο παραλήπτης να λάβει όλα τα πακέτα;

➤ Το ίδιο, αλλά αντί για GB3 έχουμε selective repeat με μέγεθος παραθύρου 3.

# Πρόβλημα 6

Υποθέστε ότι οι τιμές του timeout για το TCP είναι αρκετά μεγάλες, ώστε 5 διαδοχικά segments δεδομένων και τα αντίστοιχα ACKs μπορούν να αποσταλούν μεταξύ των τερματικών A (αποστολέας) και B (παραλήπτης). Υποθέστε ότι το A στέλνει 5 segments δεδομένων στο B και το δεύτερο segment χάνεται.

➤ Πόσα segments στέλνει συνολικά ο A και πόσα ACKs στέλνει συνολικά ο B και ποιοί είναι οι αριθμοί ακολουθίας τους;

Απάντηση:

Ο A στέλνει 6 segments συνολικά. Αυτά θα είναι τα αρχικά μηνύματα 1,2,3,4,5 και αργότερα το 2.

Το B στέλνει 5 ACKs. Αυτά θα είναι 4 ACKs με αριθμό ακολουθίας 2 και ένα ACK αριθμό ακολουθίας 6.