

# TCP protocol

---

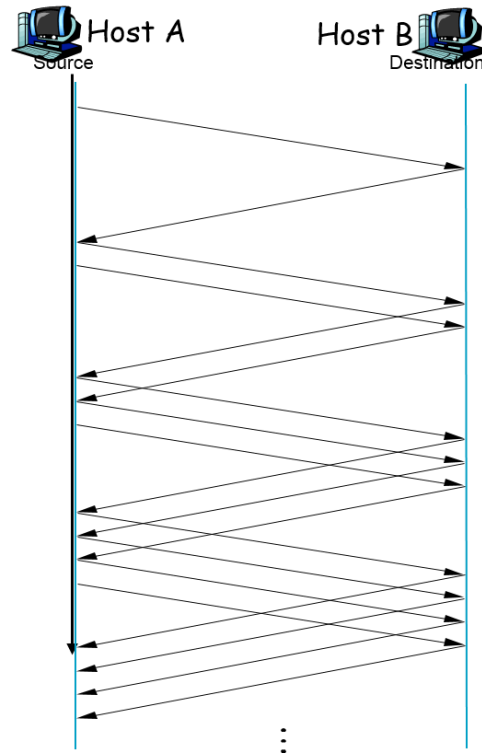
# Άσκηση 1

---

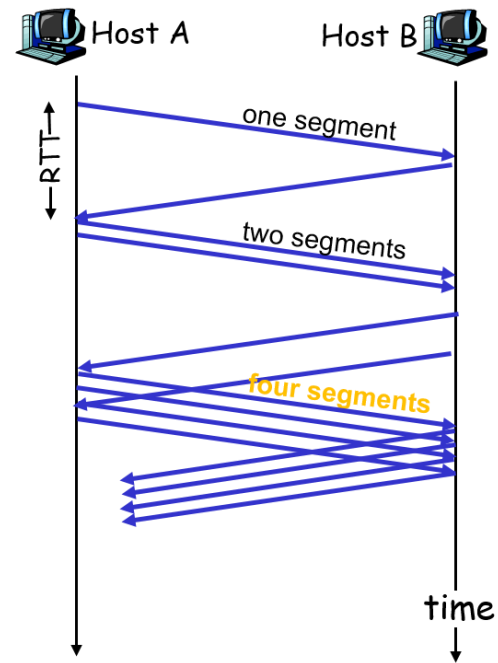
Είναι το ίδιο να αυξάνεται το congestion window κατά μία μονάδα μετά τη λήψη από κάθε ACK πακέτου με το να αυξάνεται σε κάθε RTT; Αν δεν είναι το ίδιο σε ποια περίπτωση επιτυγχάνεται μεγαλύτερο data-rate στην TCP σύνδεσή; Δώστε σχήμα.

# Άσκηση 1

## Additive Increase



## Exponential Increase



# Άσκηση 1

---

Αύξηση του παραθύρου συμφόρησης (congestion window) κατά μία μονάδα σε κάθε RTT σημαίνει γραμμική αύξησή του παραθύρου στον χρόνο (αριστερά στο σχήμα). Αντίθετα, αύξηση του παραθύρου κατά μία μονάδα με την λήψη κάθε ACK πακέτου ισοδυναμεί με διπλασιασμό του παραθύρου σε κάθε RTT, δηλαδή εκθετική αύξησή του στον χρόνο (δεξιά στο σχήμα). Στην δεύτερη περίπτωση το μέγεθος του παραθύρου αυξάνεται πολύ γρήγορα, επιτυγχάνοντας σύντομα μεγάλο data rate.

Στο πρωτόκολλο TCP υλοποιούνται και οι δύο αλγόριθμοι και χρησιμοποιούνται σε διαφορετικές φάσεις λειτουργίας του. Στην φάση της αργής εκκίνησης το μέγεθος του παραθύρου συμφόρησης τίθεται σε 1 MSS (Maximum Segment Size) και στην συνέχεια αυξάνεται εκθετικά στον χρόνο. Στην φάση της αποφυγής συμφόρησης το μέγεθος του παραθύρου αυξάνεται γραμμικά στον χρόνο μέχρι να συμβεί κάποιο συμβάν απώλειας πακέτου.

# Άσκηση 2

---

Κατά τη διάρκεια μίας TCP ροής όταν συμβεί timeout τι επιπτώσεις θα υπάρχουν στο μέγεθος της πληροφορίας που θα στείλει αμέσως μετά ο TCP transmitter;

Αν συμβούν 2 duplicate ACKs αντί για timeout τι γίνεται σε αυτή την περίπτωση;

# Άσκηση 2

---

Κατά την διάρκεια μιας TCP ροής, ο TCP transmitter εκτιμά την συμφόρηση ανιχνεύοντας events απώλειας πακέτων. Ένα πακέτο θεωρείται χαμένο αν

- εκπνεύσει ο χρονομετρητής (timeout event).
- ληφθούν 3 ίδια ACKs (3 duplicate ACKs event).

Στην περίπτωση που εκπνεύσει ο χρονομετρητής, ο αποστολέας εισέρχεται στην φάση αργής εκκίνησης, δηλαδή θέτει το παράθυρο συμφόρησης σε 1 MSS και μετά αυξάνει το παράθυρο εκθετικά, μέχρι να φτάσει στο μισό της τιμής που είχε πριν από το συμβάν λήξης χρόνου ( $\text{threshold} = \text{CongWin}/2$ ). Έπειτα, ο αποστολέας εισέρχεται στην φάση αποφυγής συμφόρησης και το παράθυρο αυξάνεται γραμμικά.

# Άσκηση 2

---

Στην περίπτωση λήψης 2 duplicate ACKs, ο μετρητής των duplicate ACKs θα πάρει την τιμή 2, χωρίς όμως να πυροδοτεί κανένα από τους δύο παραπάνω τύπους event απώλειας πακέτου. Έτσι, ο αλγόριθμος αργής εκκίνησης θα συνεχίσει να εκτελείται κανονικά. Στην περίπτωση όμως που ληφθεί και τρίτο duplicate ACK, ο αποστολέας θα υποδιπλασιάσει το παράθυρο συμφόρησης, και στη συνέχεια θα εισέλθει στην φάση αποφυγής συμφόρησης, αυξάνοντας το παράθυρο γραμμικά.

# Άσκηση 3α

---

Ο host A και B επικοινωνούν μέσω μίας TCP σύνδεσης, ο host B έχει ήδη λάβει από τον A 126 bytes. Υποθέστε ότι ο A στέλνει στον B δύο segments το ένα μετά το άλλο. Το πρώτο και το δεύτερο segment περιέχουν 70 και 50 bytes δεδομένων, αντίστοιχα. Το source port number είναι 302 και το destination port number είναι 80. Ο host B στέλνει ένα acknowledgment κάθε φορά που λαμβάνει ένα τμήμα δεδομένων από τον A.

Στο δεύτερο τμήμα δεδομένων (segment) που στέλνεται από τον host A στον B, ποιος είναι ο αριθμός ακολουθίας, το source port number και το destination port number;

## Απάντηση:

Για το δεύτερο τμήμα δεδομένων που στέλνεται από τον host A στον B, ο αριθμός ακολουθίας είναι  $126 + 70 + 1 = 197$ .

Το source port number είναι 302 και το destination port number είναι 80.



# Άσκηση 3β

---

Ο host A και B επικοινωνούν μέσω μίας TCP σύνδεσης, ο host B έχει ήδη λάβει από τον A 126 bytes. Υποθέστε ότι ο A στέλνει στον B δύο segments το ένα μετά το άλλο. Το πρώτο και το δεύτερο segment περιέχουν 70 και 50 bytes δεδομένων, αντίστοιχα. Το source port number είναι 302 και το destination port number είναι 80. Ο host B στέλνει ένα acknowledgment κάθε φορά που λαμβάνει ένα τμήμα δεδομένων από τον A.

Ας υποθέσουμε ότι το πρώτο τμήμα δεδομένων φτάνει στον A πριν από το δεύτερο. Ποιος θα είναι ο αριθμός acknowledgment, το source port number και το destination port number που στέλνει ο A στον B;

Απάντηση

Ο αριθμός acknowledgment θα είναι ο  $126 + 70 + 1 = 197$

Το source port number είναι 80 και το destination port number είναι 302.

# Άσκηση 3γ

---

Ο host A και B επικοινωνούν μέσω μίας TCP σύνδεσης, ο host B έχει ήδη λάβει από τον A 126 bytes. Υποθέστε ότι ο A στέλνει στον B δύο segments το ένα μετά το άλλο. Το πρώτο και το δεύτερο segment περιέχουν 70 και 50 bytes δεδομένων, αντίστοιχα. Το source port number είναι 302 και το destination port number είναι 80. Ο host B στέλνει ένα acknowledgment κάθε φορά που λαμβάνει ένα τμήμα δεδομένων από τον A.

Ας υποθέσουμε ότι το δεύτερο τμήμα δεδομένων φτάνει στον A πριν από το δεύτερο. Ποιος θα είναι ο αριθμός acknowledgment, το source port number και το destination port number που στέλνει ο A στον B;

Απάντηση

Ο αριθμός acknowledgment θα είναι ο 127

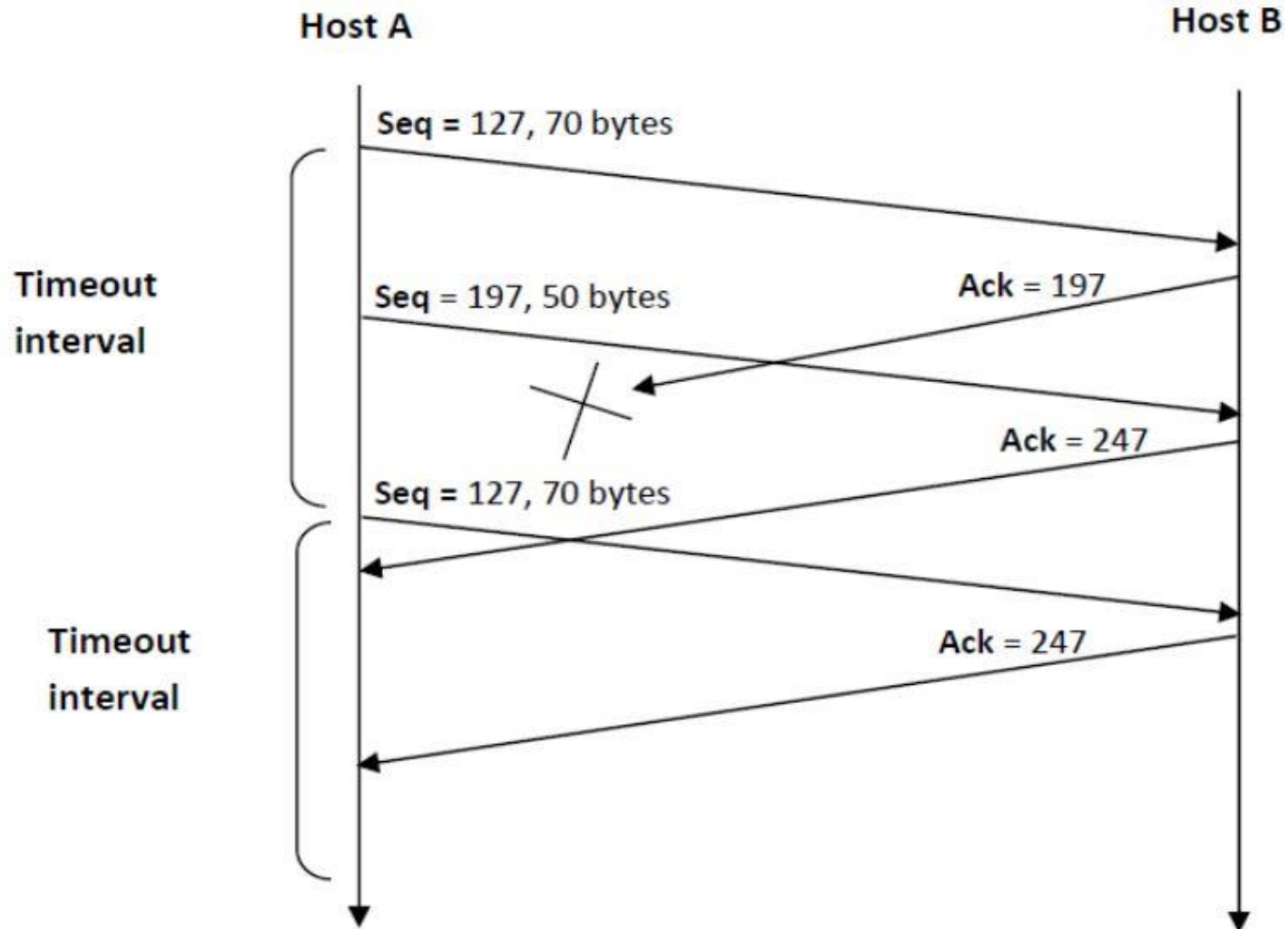
Το source port number είναι 80 και το destination port number είναι 302.

# Άσκηση 3δ

---

d) Υποθέστε ότι τα δύο τμήματα δεδομένων φτάνουν με τη σειρά τους από τον A στον B. Θεωρείστε ότι το πρώτο acknowledgment χάνεται ενώ το δεύτερο φτάνει αφού λήξει το πρώτο time-out. Σχεδιάστε ένα διάγραμμα που να δείχνει αυτά τα τμήματα δεδομένων και όλα τα υπόλοιπα καθώς και τα acknowledgments. (Μπορείτε να υποθέσετε ότι δεν υπάρχει απώλεια κάποιου άλλου πακέτου.) Για κάθε ένα τμήμα δεδομένων στο διάγραμμα σας να εμφανίζεται ο αριθμός ακολουθίας και ο αριθμός των bytes δεδομένων. Επίσης για κάθε acknowledgment δώστε τον αριθμό acknowledgment.

# Άσκηση 3δ



# Άσκηση 4

---

Θεωρείστε ότι μία και μοναδική TCP σύνδεση χρησιμοποιεί ένα link των 10 Mbps, το οποίο δεν κάνει buffer τα δεδομένα. Υποθέστε ότι ο αποστολέας στέλνει ένα μεγάλο αρχείο στον παραλήπτη, ο παραλήπτης έχει buffer μεγαλύτερο από το μέγεθος του αρχείου. Θεωρείστε τα εξής: MSS = 1500 bytes, RTT = 100 msec και η TCP σύνδεση είναι σε φάση αποφυγής συμφόρησης.

Ποιο είναι το μέγιστο μέγεθος παραθύρου (σε segments) που μπορεί να επιτύχει αυτή η σύνδεση TCP.

# Άσκηση 4

---

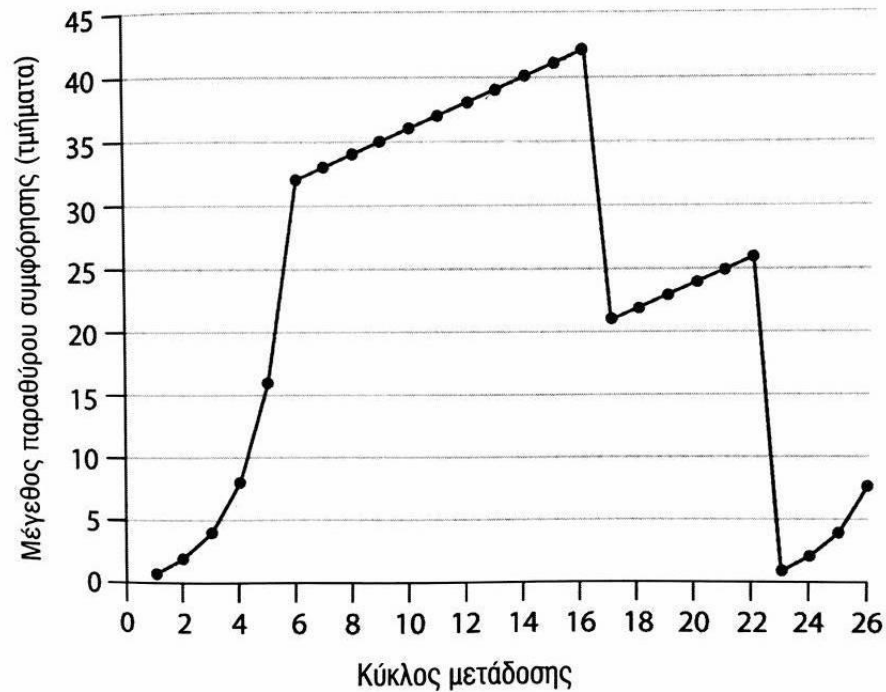
Στον μηχανισμό ελέγχου συμφόρησης του TCP, ο ρυθμός με τον οποίο ο αποστολέας στέλνει δεδομένα εξαρτάται από δύο μεταβλητές, το MSS, και το μέγεθος του παραθύρου συμφόρησης. Συγκεκριμένα ο ρυθμαπόδοση (throughput) του αποστολέα είναι  $W * MSS / RTT$ .

Όπου  $W$  είναι το μέγιστο μέγεθος παραθύρου σε segments, τότε ισχύει ότι το

$$W * MSS / RTT = 10\text{Mbps} \Rightarrow W * 1500 / 100 * 10^{-3} = 10 * 10^6$$

Άρα λύνουμε ως προς  $W$  για να βρούμε το μέγιστο αριθμό σε segments.

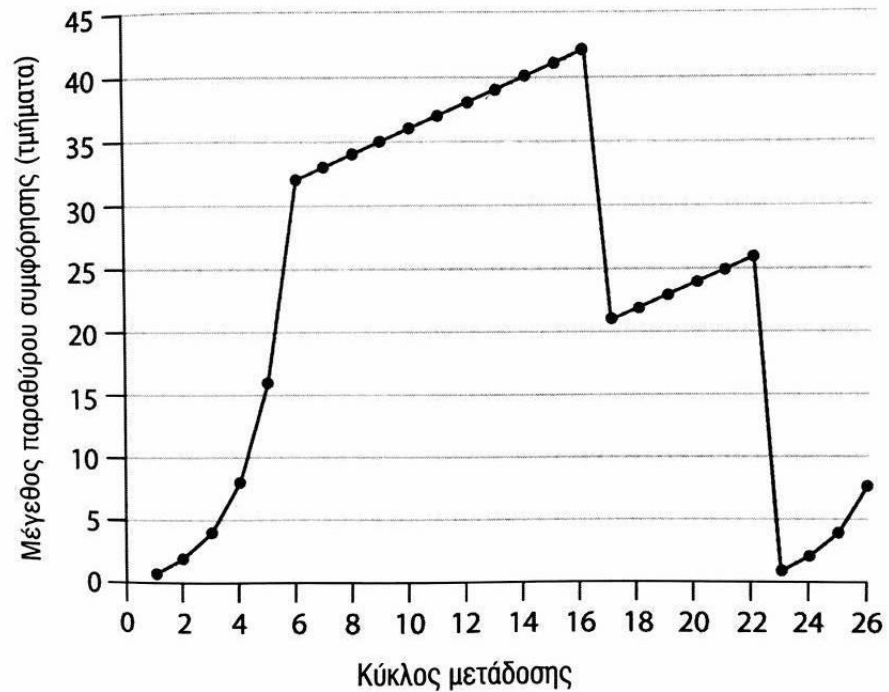
# Άσκηση 5



α. Καθορίστε τα χρονικά διαστήματα όπου λειτουργεί η αργή εκκίνηση TCP (slow start).

β. Καθορίστε τα χρονικά διαστήματα όπου λειτουργεί η αποφυγή συμφόρησης TCP (congestion avoidance).

# Άσκηση 5



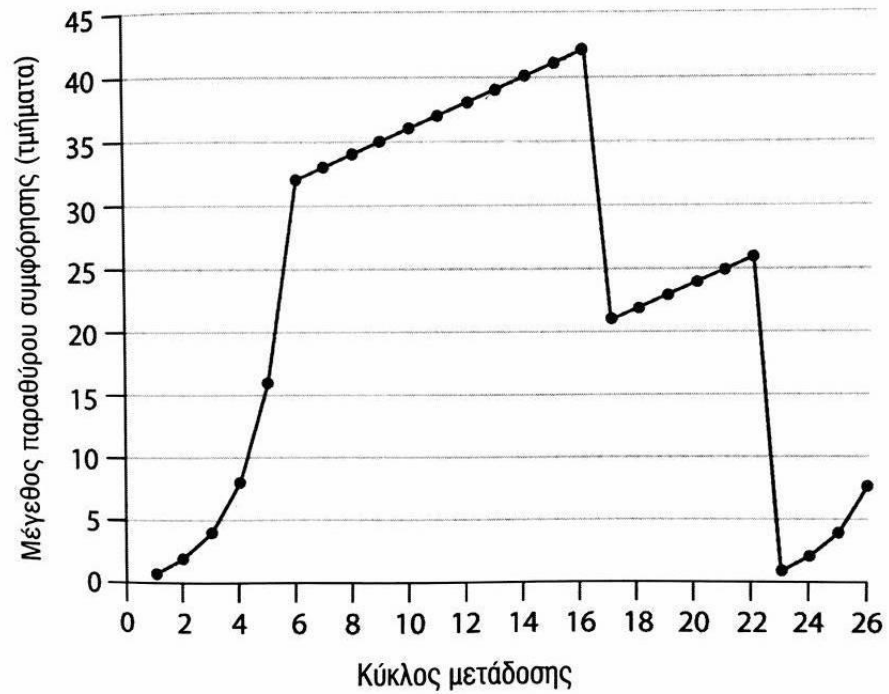
γ. Μετά από τον 16ο κύκλο μετάδοσης, τί είδους απώλεια πακέτου έχουμε;

δ. Μετά από τον 22ο κύκλο μετάδοσης, τί είδους απώλεια πακέτου έχουμε;



# Άσκηση 5

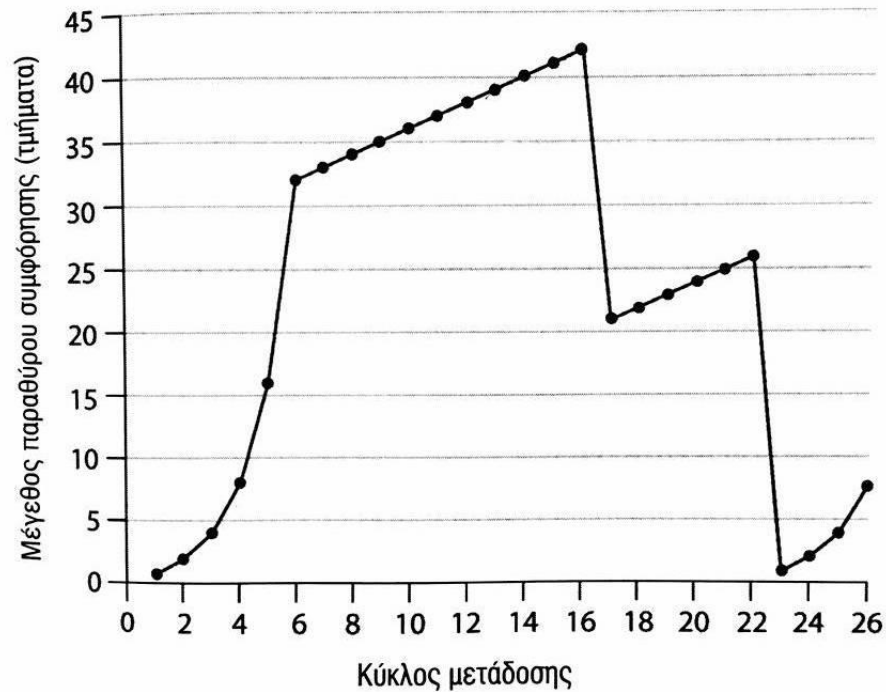
---



ε. Ποιά είναι η τιμή της threshold στον 18ο γύρο μετάδοσης;

ζ. Ποιά είναι η τιμή του threshold στον 24ο γύρο μετάδοσης;

# Άσκηση 5



η. Κατά τη διάρκεια ποιού γύρου στέλνεται το 70ο τμήμα TCP;

θ. Υποθέστε ότι απώλεια πακέτου συμβαίνει ανιχνεύεται μετά τον 26ο γύρο από την λήψη ενός **triple duplicate ACK**. Ποιές είναι οι τιμές του congestion window και threshold;

# Άσκηση 5

---

A)  $t=1$  εως  $t=6$

B)  $t=6$  εως  $t=16$  και  $t=17$  εως  $t=22$

Γ) 3 duplicates ACKs

Δ) timeout event

E) threshold = 21

Z) threshold = 13

H) Στον 7<sup>ο</sup> γύρο

Θ)  $CongWin = 8/2 = 4$ , threshold = 4

# Άσκηση 6α

---

Θεωρείστε μία σύνδεση TCP η οποία στέλνει 12 πακέτα. Κατά τη διάρκεια της μετάδοσης, το 4ο πακέτο χάνεται λόγω συμφόρησης. Υποθέστε ότι η **επαναμετάδοση** του πακέτου είναι **επιτυχής** και ότι κανένα άλλο πακέτο δεν χάνεται. Επίσης θεωρείστε ότι:

RTT = 1 sec

TCP retransmit timeout counter = 2 sec

Το πρώτο πακέτο στέλνεται την  $t = 0$

**Χωρίς** fast retransmit και **χωρίς** fast recovery.

# Άσκηση 6α

---

Χρονική στιγμή t	Φάση του TCP (SS / AIMD)	Αριθμός πακέτων που στέλνονται	Threshold	Πακέτα προς μετάδοση
0	SS	1	INF	1
1	SS	2	INF	2,3
2	SS	4	INF	4,5,6,7
3	Πάγωμα του TCP για ένα RTT			
4	SS	1	2	4
5	SS	2	2	8,9
6	AIMD	3	2	10,11,12

# Άσκηση 6β

---

Θεωρείστε μία σύνδεση TCP η οποία στέλνει 12 πακέτα. Κατά τη διάρκεια της μετάδοσης, το 4ο πακέτο χάνεται λόγω συμφόρησης. Υποθέστε ότι η **επαναμετάδοση** του πακέτου είναι **επιτυχής** και ότι κανένα άλλο πακέτο δεν χάνεται. Επίσης θεωρείστε ότι:

RTT = 1 sec

TCP retransmit timeout counter = 2

Το πρώτο πακέτο στέλνεται την  $t = 0$

**Έχουμε** fast retransmit , αλλά **όχι** fast recovery.

# Άσκηση 6β

---

Χρονική στιγμή t	Φάση του TCP (SS / AIMD)	Αριθμός πακέτων που στέλνονται	Threshold	Πακέτα προς μετάδοση
0	SS	1	INF	1
1	SS	2	INF	2,3
2	SS	4	INF	4,5,6,7
3	SS	1	2	4
4	SS	2	2	8,9
6	AIMD	3	2	10,11,12