

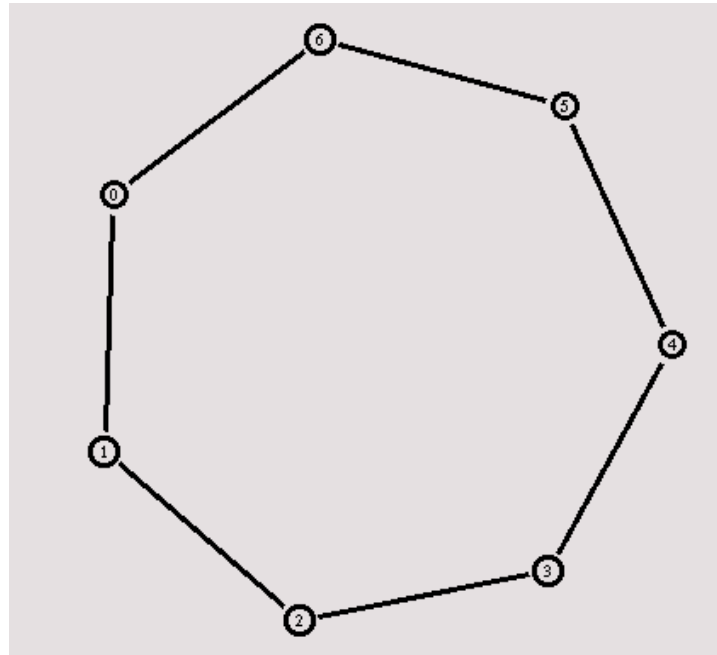
Προσομοίωση Δικτύων

4η Άσκηση

Σύνθετες τοπολογίες, διακοπή συνδέσεων,
δυναμική δρομολόγηση

Δημιουργία multihop τοπολογίας

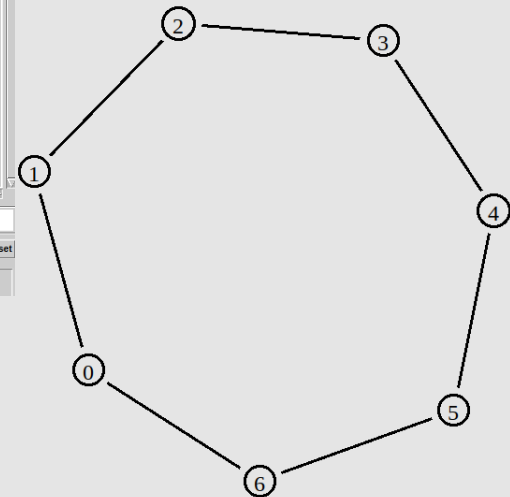
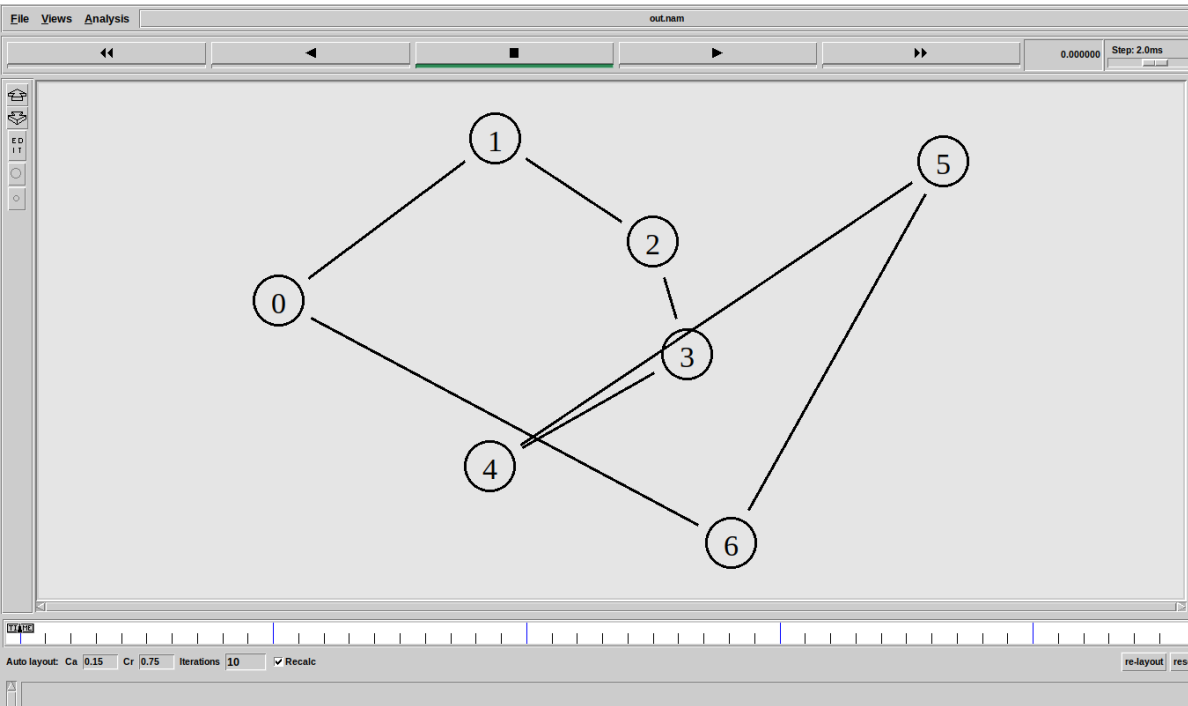
- Υλοποιήστε την παρακάτω τοπολογία βασιζόμενη σε προηγούμενα έτοιμα σενάρια



- Όλα τα links είναι 1Mb bandwidth, 10ms delay και DropTail
- Ένα data flow from $n(0)$ to $n(3)$
- CBR, 500byte packets, 0.005sec interval

Δημιουργία multihop τοπολογίας

- Σημείωση, αν το δίκτυό σας εμφανίζεται μπερδεμένο όπως το παρακάτω πατήστε το κουμπί «re-layout» στο κάτω δεξιό μέρος της οθόνης

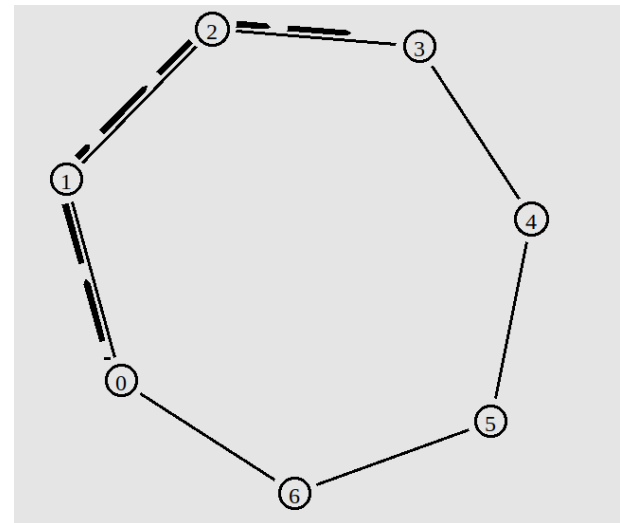
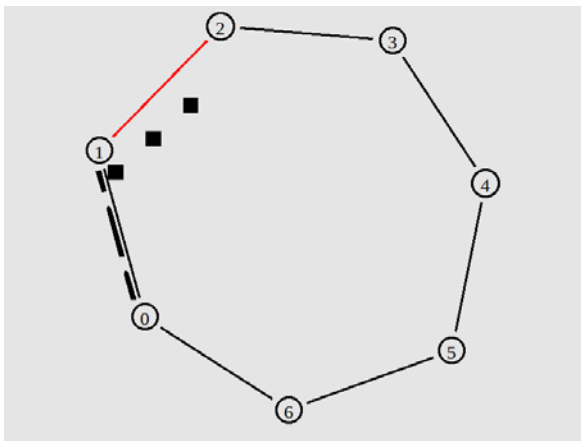


Δημιουργία multihop τοπολογίας

- Βασιζόμενοι στο NAM πόσες προωθήσεις γίνονται για κάθε πακέτο;
- Χρησιμοποιήστε το “4th_delay_1” awk script για να αξιολογήσετε την απόδοση του δικτύου και καταγράψτε της
- Με βάση πιο κριτήριο γίνεται η δρομολόγηση;
 - Αλλάξτε το κόμβο λήψης για να δείτε την επίδραση της δρομολόγησης
- Ποια η καθυστέρηση ανά hop; (για το λόγο αυτό κάνετε τις παρακάτω κινήσεις)
 - Set n(6) as the receiver at tcl script
 - Set node 6 as the receiver at awk script at line marked with # #EDO
ALLAZETE TON RECEIVER
- Ορίστε πάλι τον n(3) σαν λήπτη στο tcl script
 - Με βάση το trace file αναγνωρίστε μια πλήρη ακολουθία hops

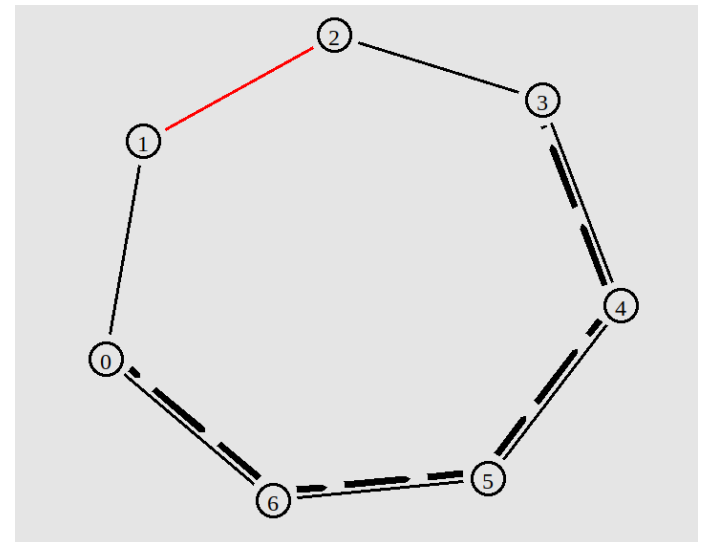
Δημιουργία Δυναμικής Τοπολογίας

- Ορίστε $n(1)$ - $n(2)$ link βλάβη χρησιμοποιώντας τις εντολές
 - `$ns rtmodel-at 1.0 down $n(1) $n(2)`
 - `$ns rtmodel-at 2.0 up $n(1) $n(2)`
 - Οι εντολές αυτές να είναι μετά τις εντολές δήλωσης κόμβων
- Οπτική παρατήρηση βλάβης
- Χρησιμοποιείτε το “4th_delay_1” για αξιολόγηση απόδοσης
- Πόσα πακέτα μεταδόθηκαν επιτυχώς;



Δημιουργία Δυναμικής Τοπολογίας

- Ενεργοποίηση δυναμικής δρομολόγησης
 - `$ns rtproto DV`
 - Η εντολή αυτή πρέπει να είναι μετά την «`set ns [new Simulator]`»
- Δείτε πως μειώνεται το πρόβλημα με παρατήρηση μέσω NAM
- Πόσα πακέτα σώθηκαν λόγω του DV πρωτοκόλλου δρομολόγησης?
- **Σημείωση:** Απαντήστε με βάση τα επιτυχώς μεταδοθέντα πακέτα όχι τα dropped !!!



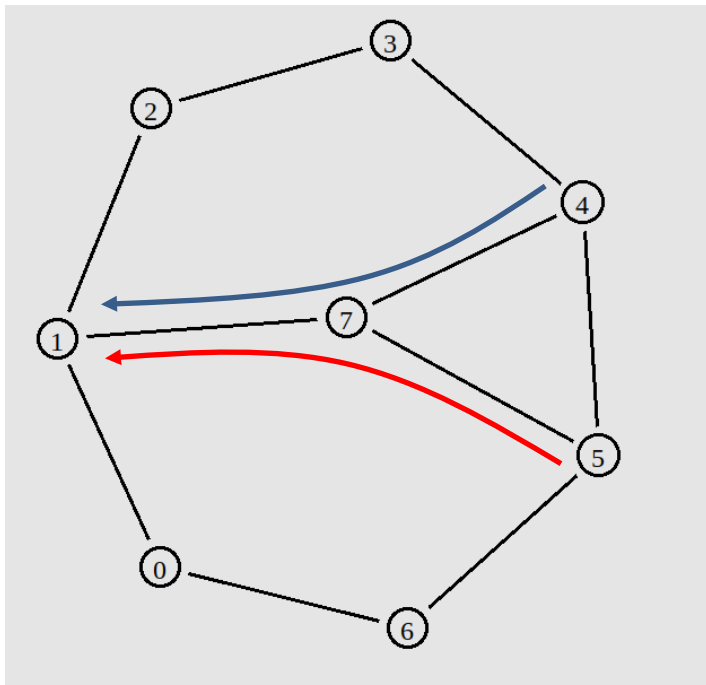
Δημιουργία Δυναμικής Τοπολογίας

- Το DV routing βασίζεται στην εύρεση της διαδρομής μικρότερου βάρους/κόστους
 - Εξ ορισμού κάθε link έχει βάρος “1”
 - Μπορούμε να ορίσουμε το βάρος κάθε σύνδεσης με την παρακάτω εντολή
 - `$ns cost $n(1) $n(2) 10`
- Χρησιμοποιείστε την παραπάνω εντολή έτσι ώστε τα πακέτα της ροής 0->3 να ακολουθούν την διαδρομή 0→6→5→4→3
- Τροποποιείστε το tcl script έτσι ώστε
 - Για το διάστημα 0.5 – 3sec τα πακέτα ακολουθούν την σύντομη διαδρομή
 - Για το διάστημα 3 – 4.5sec τα πακέτα ακολουθούν την μεγαλύτερη διαδρομή

ΑΝΑΦΟΡΑ 1

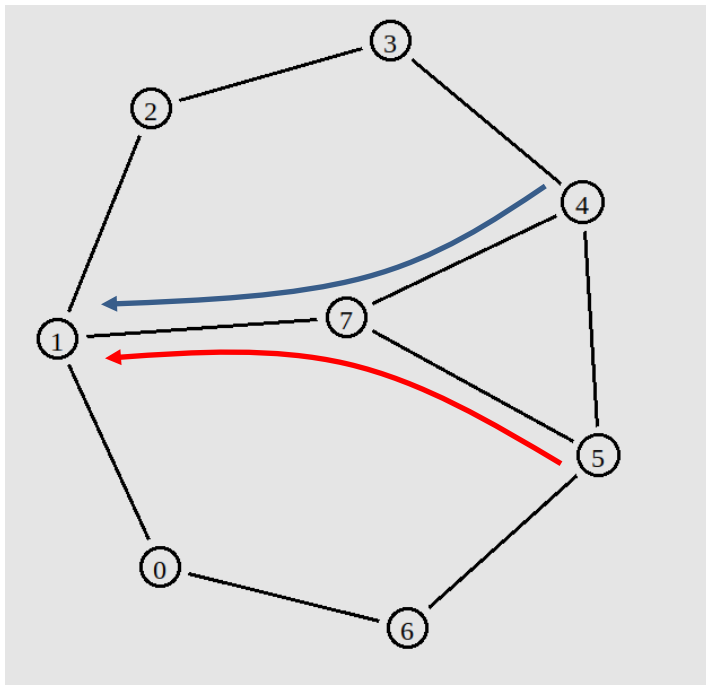
- Ελέγξατε αν η εντολή « $\$ns\ cost\ \$n(1)\ \$n(2)\ 10$ » είναι δικατευθυντήρια, δηλαδή έχει το ίδιο αποτέλεσμα στο κόστος της ζεύξης $n2-n1$ που έχει στην ζεύξη $n1-n2$.
- Με βάση την τοπολογία στην διαφάνεια 2, ορίστε μια ροή $n1-n4$ που θα περνάει από $n1-n2-n3-n4$ και μια ροή $n4-n1$ που θα περνάει από $n4-n5-n6-n0-n1$
- Με βάση το out.tr αρχείο, εντοπίστε και αντιγράψτε στην αναφορά σας μια ακολουθία γραμμών στην οποία να φαίνεται η διαδρομή ενός συγκεκριμένου πακέτου της ροής $n4-n1$

ΑΝΑΦΟΡΑ 2



- Υλοποιήστε την παραπάνω τοπολογία και τις δύο ροές πακέτων (δεν χρειάζονται χρώματα)
- CBR, 500byte packets, 0.005sec interval και για τις δύο ροές
- Εκτιμήστε αν θα υπάρξει απώλεια πακέτων μέσω NAM
- Υπολογίστε καθυστέρηση και απώλειες πακέτων μέσω 4th_delay_2.txt
 - Δεν αλλάζετε κάτι στο script

ΑΝΑΦΟΡΑ 3



- Χρησιμοποιήστε τις εντολές αλλαγής βάρους μιας ζεύξης για να αποσυμφορηθεί το δίκτυο.
- Χρησιμοποιήστε τις εντολές πτώσης ζεύξης για να αποσυμφορηθεί το δίκτυο.