3η Σειρά Γραπτών Ασκήσεων

Αλγόριθμοι και Πολυπλοκότητα

ΣΗΜΜΥ, Εθνικό Μετσόβιο Πολυτεχνείο

Άσκηση 1 (α): Πομποί και Δέκτες

Είσοδος: Ακολουθία $a = a_1, \ldots, a_n$ όπου $\forall i \; a_i = r$ ή $a_i = t$.

Ζητείται: Μέγιστη υπακολουθία όπου κάθε πομπός αντιστοιχεί σε ένα δέκτη προς τα δεξιά του.

Άσκηση 1 (α): Πομποί και Δέκτες

Λύση: Με χρήση στοίβας

- Διατρέχουμε την είσοδο από αριστερά προς τα δεξιά, έχοντας αρχικοποιήσει ένα μετρητή ζευγών $p = 0$.

Άσκηση 1 (α): Πομποί και Δέκτες

Λύση: Με χρήση στοίβας

- Διατρέχουμε την είσοδο από αριστερά προς τα δεξιά, έχοντας αρχικοποιήσει ένα μετρητή ζευγών $p = 0$.
- Κάθε φορά που συναντάμε πομπό, τον βάζουμε στη στοίβα.

Λύση: Με χρήση στοίβας

- Διατρέχουμε την είσοδο από αριστερά προς τα δεξιά, έχοντας αρχικοποιήσει ένα μετρητή ζευγών $p = 0$.
- Κάθε φορά που συναντάμε πομπό, τον βάζουμε στη στοίβα.
- Κάθε φορά που συναντάμε δέκτη, αν η στοίβα είναι μη κενή βγάζουμε έναν πομπό και αυξάνουμε το μετρητή $p++$, διαφορετικά προχωράμε στο επόμενο στοιχείο.

Άσκηση 1 (α): Πομποί και Δέκτες

Λύση: Με χρήση στοίβας

- Διατρέχουμε την είσοδο από αριστερά προς τα δεξιά, έχοντας αρχικοποιήσει ένα μετρητή ζευγών $p = 0$.
- Κάθε φορά που συναντάμε πομπό, τον βάζουμε στη στοίβα.
- Κάθε φορά που συναντάμε δέκτη, αν η στοίβα είναι μη κενή βγάζουμε έναν πομπό και αυξάνουμε το μετρητή $p++$, διαφορετικά προχωράμε στο επόμενο στοιχείο.
- Μήκος μέγιστης ζητούμενης υπακολουθίας: $2p$.

Πολυπλοκότητα

$$O(n)$$

Άσκηση 1 (β): Πομποί και Δέκτες

Είσοδος: Ακολουθία n (άρτιος) ζευγών (T_i, R_i), όπου T_i η κατανάλωση ισχύος της κεραίας i όταν λειτουργεί ως πομπός και $R_i \leq T_i$ όταν λειτουργεί ως δέκτης.

Ζητείται: Ελάχιστη συνολική κατανάλωση ισχύος, όπου για κάθε i που λειτουργεί ως πομπός υπάρχει ένα και μοναδικό $j > i$ που λειτουργεί ως δέκτης.

Άσκηση 1 (β): Πομποί και Δέκτες

Λύση: Δυναμικός Προγραμματισμός

Αναδρομική Σχέση

$$C[i, j] = \min\{C[i-1, j+1] + R_i, C[i-1, j-1] + T_i\}$$

όπου $C[i, j]$ το κόστος της βέλτιστης λύσης μέχρι και την i-οστή κεραία, έχοντας $j \leq i$ πομπούς που δεν έχουν αντιστοιχηθεί σε δέκτες προς το παρόν.

Άσκηση 1 (β): Πομποί και Δέκτες

Λύση:

$$C[i,j] = \min\{C[i-1,j+1] + R_i, C[i-1,j-1] + T_i\}$$

Αν έχουμε τη βέλτιστη λύση για $i-1$ κεραίες τότε:

- Είτε υπάρχουν $j+1$ πομποί που δεν έχουν αντιστοιχηθεί σε δέκτες και η i-οστή κεραία γίνεται δέκτης οπότε μένουν j πομποί που δεν έχουν αντιστοιχηθεί σε δέκτες και προστίθεται R_i στη συνολική κατανάλωση ισχύος.
- Είτε υπάρχουν $j-1$ πομποί που δεν έχουν αντιστοιχηθεί σε δέκτες και η i-οστή κεραία γίνεται πομπός οπότε μένουν j πομποί που δεν έχουν αντιστοιχηθεί σε δέκτες και προστίθεται T_i στη συνολική κατανάλωση ισχύος.

Άσκηση 1 (β): Πομποί και Δέκτες

Λύση:
Η ζητούμενη λύση είναι η $C[n,0]$.

Πολυπλοκότητα

Υπολογισμός ενός εκ των $\Theta(n^2/2)$ στοιχείων του C σε $O(1) \implies$

$$O(n^2)$$

Άσκηση 1 (β): Πομποί και Δέκτες

Λύση:
Η ζητούμενη λύση είναι η $C[n, 0]$.

Πολυπλοκότητα

Υπολογισμός ενός εκ των $\Theta(n^2/2)$ στοιχείων του C σε $O(1) \Longrightarrow$

$$O(n^2)$$

Υπάρχει πιο αποδοτικός αλγόριθμος?

Άσκηση 1 (β): Πομποί και Δέκτες

Καλύτερη Λύση: Για κάποια ανάθεση πομπών-δεκτών έστω $S[i]$ το πλήθος πομπών μείον το πλήθος δεκτών στο διάστημα $[1, i]$. Μία ανάθεση είναι έγκυρη αν και μόνο αν για κάθε $i \in \{1, \ldots, n\}$ ισχύει $S[i] \geq 0$ και $S[n] = 0$.

Θεωρούμε ότι έχουμε μια βέλτιστη έγκυρη ανάθεση στο διάστημα $[1, i]$. Αν ο i είναι περιττός, για να την επεκτείνουμε βάζουμε ένα δέκτη στο $i + 1$ ($S[i + 1] = 0$). Αν είναι άρτιος, βάζουμε ένα δέκτη στο $i + 1$, αλλά έχουμε $S[i + 1] = -1$, άρα κάποιος δέκτης στο $[1, i + 1]$ πρέπει να μετατραπεί σε πομπό: Μετατρέπουμε αυτόν που ελαχιστοποιεί την κατανάλωση, δηλαδή αυτόν με το ελάχιστο $T_j - R_j$. Τώρα έχουμε $S[i + 1] = 0$.

Για να βρίσκουμε το δέκτη με το ελάχιστο $T_j - R_j$ θα χρειαστούμε ένα min-heap στο οποίο οι λειτουργίες εισαγωγής και εξαγωγής ελαχίστου στοιχείου γίνονται το πολύ σε $O(\log n)$.

Άσκηση 1 (β): Πομποί και Δέκτες

Αλγόριθμος:

- Αρχικοποίησε ένα κενό min-heap Q
- Για $i = 1 \ldots n$
  - Θεώρησε τον i ως δέκτη, πρόσθεσε R_i στο συνολικό κόστος και βάλε το ζεύγος $(T_i - R_i, i)$ στο Q
  - Αν $S[i] < 0$, αφαίρεσε από το Q το δέκτη με το ελάχιστο $T_j - R_j$, μετάτρεψέ τον σε πομπό και αύξησε το συνολικό κόστος κατά $T_j - R_j$.

Χρονική Πολυπλοκότητα

$$O(n \log n)$$

Μένει να δείξουμε το επαγωγικό βήμα, δηλαδή ότι η λύση στο διάστημα $[1, i+1]$ είναι βέλτιστη, δεδομένου ότι η λύση στο $[1, i]$ είναι βέλτιστη.

- Αν i περιττός ($S[i] = 1$), τότε προφανώς η θέση $i+1$ πρέπει να έχει δέκτη για να είναι έγκυρη η λύση και γιατί είναι το φθηνότερο. Άρα η λύση μας είναι βέλτιστη.
- Αν i άρτιος ($S[i] = 0$) τότε αν προσθέσουμε δέκτη στη θέση $i+1$ για να είναι έγκυρη η λύση πρέπει να μετατρέψουμε ένα δέκτη σε πομπό. Προφανώς αφού η λύση είναι μέχρι στιγμής ελαχίστου κόστους, προσθέτοντας το μικρότερο $T_j - R_j$ λόγω της αλλαγής (και R_{i+1} αντί T_{i+1}) έχουμε λύση μικρότερου δυνατού κόστους άρα ακόμα βέλτιστη.

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$								
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 0

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	-1							
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 0 + 6 = 6

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1							
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 6 + (8-6) = 8

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	0						
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 8 + 3 = 11

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	0	-1					
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 11 + 1 = 12

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	0	1					
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 12 + (5-1)=16

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	0	1	0				
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: $16 + 4 = 20$

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	0	1	0	-1			
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 20 + 3 = 23

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	0	1	2	1			
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 23 + (6-4)=25

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	0	1	2	1	0		
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 25+7=32

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	0	1	2	1	0	-1	
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 32+5=37

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	2	3	4	3	2	1	
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 37+(9-3)=43

Άσκηση 1 (β): Πομποί και Δέκτες

Παράδειγμα:

i	1	2	3	4	5	6	7	8
$S(i)$	1	2	3	4	3	2	1	0
T_i	8	9	5	6	10	30	12	2
R_i	6	3	1	4	3	7	5	1

Κατανάλωση: 43+1=44

Άσκηση 2: Διακοπές στην Ικαρία

Είσοδος: Πλειάδες με 3 στοιχεία, όπου το $s_i < f_i$ και a_i η προσφορά για αυτές τις μέρες
Έξοδος: Το μέγιστο ποσό που μπορεί να αποκομίσει ο κύριος Μάκης από τις προσφορές που του έχουν γίνει για αυτές τις n μέρες
Περιορισμός: Δε γίνεται να αποδεχτεί διαφορετικές προσφορές από 2 ή περισσότερους ενοικιαστές για κάποια μέρα

Θα μοντελοποιήσουμε το πρόβλημα ως DAG:

1. Φτιάχνουμε έναν κόμβο για καθεμία από τις n μέρες
2. Για κάθε προσφορά φτιάχνουμε μια κατευθυνόμενη ακμή από τη μέρα s_i στην f_i με βάρος a_i
3. Στη συνέχεια πρέπει να προσθέσουμε ακμές μηδενικού βάρους από κάθε μέρα i στη i+1, όταν δεν έχει ήδη τοποθετηθεί ακμή. Έτσι καλύπτουμε τις μέρες όπου πιθανώς το σπίτι δε θα ενοικιαστεί. Αυτό είναι απαραίτητο για να βρούμε στη συνέχεια τη βέλτιστη λύση αλλά και για να είμαστε βέβαιοι ότι ο γράφος που έχουμε κατασκευάσει είναι συνεκτικός

Άσκηση 2: Διακοπές στην Ικαρία

- Είμαστε βέβαιοι ότι έχουμε κατασκευάσει ένα κατευθυνόμενο ακυκλικό γράφημα, αφού από τη δομή του προβλήματος δε γίνεται να εμφανιστούν "πίσω" ακμές
- Για να βρούμε το μέγιστο κέρδος πρέπει να βρούμε το μακρύτερο μονοπάτι σε DAG. Αυτό μπορεί να γίνει με αποδοτικό τρόπο αν βάλουμε ένα μείον μπροστά από το κάθε βάρος και τρέξουμε το shortest path algorithm σε DAG.

Άσκηση 2: Διακοπές στην Ικαρία

- Είμαστε βέβαιοι ότι έχουμε κατασκευάσει ένα κατευθυνόμενο ακυκλικό γράφημα, αφού από τη δομή του προβλήματος δε γίνεται να εμφανιστούν "πίσω" ακμές
- Για να βρούμε το μέγιστο κέρδος πρέπει να βρούμε το μακρύτερο μονοπάτι σε DAG. Αυτό μπορεί να γίνει με αποδοτικό τρόπο αν βάλουμε ένα μείον μπροστά από το κάθε βάρος και τρέξουμε το shortest path algorithm σε DAG.

Πολυπλοκότητα:
Έχουμε n κορυφές και O(m+n) ακμές, λόγω της κατασκευής. Για να βρούμε το longest path χρειαζόμαστε O(V+E)=O(m+n) χρόνο. Η κατασκευή παίρνει επίσης O(m+n). Άρα συνολικά O(m+n).

Άσκηση 3: Επιστροφή στη Γη

Θα δώσουμε δύο διαφορετικές λύσεις, μία για όσους έλυσαν την άσκηση για μονοπάτια(στα οποία επιτρέπεται η επανάληψη ακμών και κορυφών) και μία για όσους την έλυσαν μόνο για απλά μονοπάτια(διαδρομή η οποία περνάει από διαφορετικές κορυφές).

Walks(η επανάληψη επιτρέπεται):

- Στόχος είναι να κατασκευάσουμε ένα καινούριο γράφο G'(V',E') ο οποίος θα περιέχει όλες τις επιτρεπτές κινήσεις.
- Μετά μπορούμε να τρέξουμε BFS στο G' από τον s και να ανακτήσουμε τη συντομότερη διαδρομή από τον s στον t(αν υπάρχει) με πολυπλοκότητα O(V+E')=O(V^2).

Άσκηση 3: Επιστροφή στη Γη

Κατασκευή του G':

Πρώτη Λύση:

- Τροποποίηση BFS
- Για κάθε κορυφή v φτιάξε το σύνολο $N_1(v)$, το οποίο αποτελείται απο τους γείτονες της v. Αυτό θα πάρει O(V) χρόνο.
- Για το $N_2(v)$, δηλαδή το σύνολο που περιέχει τους κόμβους στους οποίους μπορούμε να φτάσουμε από τον v σε 2 βήματα, παίρνω την ένωση που προκύπτει από το $N_1(v)$, για κάθε κόμβο $u \epsilon N_1(v)$. Αυτό θα πάρει $O(V^2)$ χρόνο. Με παρόμοιο τρόπο κατασκευάζουμε τα $N_3(v), N_4(v), N_5(v)$.

Άσκηση 3: Επιστροφή στη Γη

- Ένα σημαντικό βήμα είναι ότι σε κάθε σύνολο πρέπει να κρατάμε μόνο μια φορά την κάθε κορυφή, αφαιρώντας τα πιθανά αντίγραφά της που έχουν προκύψει κατά την κατασκευή του συνόλου. Με αυτόν τον τρόπο το κάθε σύνολο έχει μέγεθος O(V) και έτσι εξασφαλίζουμε πολυπλοκότητα $O(V^2)$ για την κατασκευή καθενός εκ των $N_k(v)$.
- Αφού επαναλαμβάνουμε αυτή τη διαδικασία για κάθε κορυφή, η συνολική πολυπλοκότητα για την κατασκευή του G' είναι $O(V^3)$.

Κατασκευή του G':

Δεύτερη Λύση:

- Ένας πολύ απλός τρόπος είναι να φτιάξουμε τον πίνακα γειτνίασης A και να υπολογίσουμε το A^5
- Οι θέσεις με μη μηδενικά στοιχεία του A^5 αντιστοιχούν στις ακμές που πρέπει να φτιάξουμε στον G'
- Κάθε πολλαπλασιασμός πινάκων χρειάζεται $O(V^3)$ χρόνο, άρα η συνολική πολυπλοκότητα είναι και σε αυτήν την περίπτωση $O(V^3)$

Άσκηση 3: Επιστροφή στη Γη

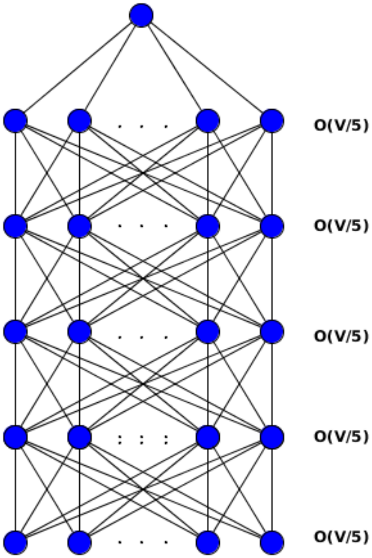
Simple Paths(η επανάληψη κορυφών δεν επιτρέπεται):

- Στη συγκεκριμένη περίπτωση δεν μπορούμε να κάνουμε κάτι καλύτερο από μια παραλλαγή του DFS για κάθε κόμβο.
- Το DFS θα φτάνει μέχρι βάθος 5 και μετά θα σταματάει. Πρέπει να ελέγχουμε για ”πίσω” ακμές. Αν βρούμε μια τέτοια, τότε δε χρειάζεται να συνεχίσουμε, γιατί ξέρουμε πως πια δεν έχουμε απλό μονοπάτι.
- Στο G’ σχηματίζουμε μια καινούρια ακμή για κάθε ζευγάρι κορυφών που απέχουν μεταξύ τους 5 βηματα(δεν είναι απαραίτητο να πρόκειται για τη συντομότερη διαδρομή μεταξύ τους)

- Άρα τελικά το DFS θα εξετάζει για κάθε κορυφή όλα τα πιθανά μονοπάτια που υπάρχουν μέχρι βάθος 5.
- Η διαδικασία αυτή θα επαναληφθεί για κάθε κορυφή.
- Θα εξηγήσουμε την πολυπλοκότητα του αλγορίθμου αναλύοντας τη συμπεριφορά του στην περίπτωση που λαμβάνουμε ως είσοδο το χειρότερο δυνατό στιγμιότυπο.

Άσκηση 3: Επιστροφή στη Γη

Το διπλανό γράφημα αποτελεί και τη χειρότερη περίπτωση όπου τα πιθανά διαφορετικά μονοπάτια ξεκινώντας από μια κορυφή είναι της τάξης του V^5. Το γράφημα έχει 5 επίπεδα και το κάθε ένα αποτελεί ένα πλήρες bipartite με το επόμενο επίπεδο. Τελικά, τρέχοντας τον αλγόριθμο από κάθε κόμβο παίρνουμε $O(V^6)$



Άσκηση 4: Διαγαλαξιακές Πτήσεις

Είσοδος: Μη κατευθυνόμενος γράφος $G(V, E, w)$, Σύνολο κορυφών R (σταθμοί επισκευής), αρχικός κόμβος ταξιδιού s

Έξοδος: Βρες όλα τα βέλτιστα μονοπάτια από τον s στους επιτρεπόμενους κόμβους-γαλαξίες

Περιορισμός: Κάθε στιγμή να έχω έναν κόμβο του R το πολύ 3 ταξίδια μακριά μου

Ιδέες?

Άσκηση 4: Διαγαλαξιακές Πτήσεις

Είσοδος: Μη κατευθυνόμενος γράφος $G(V, E, w)$, Σύνολο κορυφών R (σταθμοί επισκευής), αρχικός κόμβος ταξιδιού s

Έξοδος: Βρες όλα τα βέλτιστα μονοπάτια από τον s στους επιτρεπόμενους κόμβους-γαλαξίες

Περιορισμός: Κάθε στιγμή να έχω έναν κόμβο του R το πολύ 3 ταξίδια μακριά μου

Ιδέες?
Το πολύ 3 ταξίδια $\rightarrow$ δε μας νοιάζουν τα βάρη $\rightarrow$ BFS?

Άσκηση 4: Διαγαλαξιακές Πτήσεις

Είσοδος: Μη κατευθυνόμενος γράφος $G(V, E, w)$, Σύνολο κορυφών R (σταθμοί επισκευής), αρχικός κόμβος ταξιδιού s

Έξοδος: Βρες όλα τα βέλτιστα μονοπάτια από τον s στους επιτρεπόμενους κόμβους-γαλαξίες

Περιορισμός: Κάθε στιγμή να έχω έναν κόμβο του R το πολύ 3 ταξίδια μακριά μου

Ιδέες?
Το πολύ 3 ταξίδια $\rightarrow$ δε μας νοιάζουν τα βάρη $\rightarrow$ BFS?
Συντομότερα μονοπάτια $\rightarrow$ Dijkstra ?

Άσκηση 4: Διαγαλαξιακές Πτήσεις

Αλγόριθμος σε 2 μέρη:

1. Βρες ολους τους επιτρεπτούς κόμβους για το διαστημόπλοιο
2. Τρέξε Dijkstra για τα συντομότερα μονοπάτια μόνο με αυτούς τους κόμβους

Άσκηση 4: Διαγαλαξιακές Πτήσεις

Μέρος 1:

Θέλω να απέχω το πολύ 3 κόμβους από κάθε $v \in R$. BFS από κάθε κόμβο;

Άσκηση 4: Διαγαλαξιακές Πτήσεις

Μέρος 1:
Θέλω να απέχω το πολύ 3 κόμβους από κάθε $v \in R$. BFS από κάθε κόμβο; Όχι!
Πολυπλοκότητα στη χειρότερη περίπτωση ($|R| = |V|$) $O((V + E)V)$

Καλύτερο: Φτιάχνω ένα supernode με όλους τους κόμβους του R και τρέχω ένα BFS απο εκεί, μαρκάρω ποιές κορυφές έχω συναντήσει, πετάω τις υπόλοιπες.
Πολυπλοκότητα $O(V + E)$

Άσκηση 4: Διαγαλαξιακές Πτήσεις

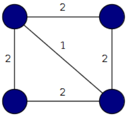
Μέρος 1:
Θέλω να απέχω το πολύ 3 κόμβους από κάθε $v \in R$. BFS από κάθε κόμβο; Όχι!
Πολυπλοκότητα στη χειρότερη περίπτωση ($|R| = |V|$) $O((V + E)V)$

Καλύτερο: Φτιάχνω ένα supernode με όλους τους κόμβους του R και τρέχω ένα BFS απο εκεί, μαρκάρω ποιές κορυφές έχω συναντήσει, πετάω τις υπόλοιπες.
Πολυπλοκότητα $O(V + E)$

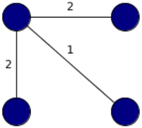
Μέρος 2:
Έχω ένα γράφημα όπου όλοι οι κόμβοι είναι επιτρεπτοί: ο Dijkstra θα μου δώσει όλα τα συντομότερα ελάχιστα μονοπάτια προς όλους τους κόμβους
Πολυπλοκότητα $O(E + V \log V)$

Άσκηση 5 (1): Απαραίτητες (και μη) ακμές

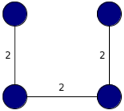
Κάθε ακμή ενός spanning tree ανήκει σε MST $\Rightarrow$ το T είναι MST ?
Όχι! Παράδειγμα:



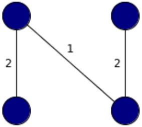
Γράφος G



MST1



ST με ακμές MST που δεν είναι MST



MST2

Άσκηση 5 (2a): Απαραίτητες (και μη) ακμές

Θα δείξω ότι:
Ακμή e απαραίτητη $\Leftrightarrow \exists$ cut $(S, V \backslash S)$: η e είναι η μοναδική ακμή ελαχίστου βάρους που τη διασχίζει

- Κατεύθυνση $\Leftarrow$: $\exists cut(S, V \backslash S)$ και $e \in \delta(S, V \backslash S)$ η μοναδική ελαφρύτερη
  - Έστω $T = \mathrm{MST}(G)$ και $T' = \mathrm{MST}(G \backslash e)$
    προφανώς $e \in T$ και $\exists e' \neq e : e' \in \delta(S, V \backslash S) \cap T'$
  - Τα MST έχουν βέλτιστες επιμέρους λύσεις:
    $w(T, S) = w(T', S)$ και $w(T, V \backslash S) = w(T', V \backslash S)$
  - $w(e') > w(e) \Rightarrow w(e') = w(e) + c, c > 0$
  - Άρα $w(T') = w(T) - w(e) + w(e) + c \Rightarrow w(T') > w(T)$

Άσκηση 5 (2a): Απαραίτητες (και μη) ακμές

- Κατεύθυνση $\Rightarrow \exists e$ που είναι απαραίτητη
  - Αν αφαιρέσουμε την ακμή e από ένα MST T που τη χρησιμοποιεί, δημιουργείται μία τομή $(S, V \setminus S)$
  - Οποιοδήποτε MST θα γεφυρώσει το S και το $V \setminus S$ (ξεχωριστά) με το ίδιο κόστος. Έστω T' ένα MST$(G \setminus e)$το οποίο είναι ίδιο με το T όσον αφορά τις συνιστώσες S, $V \setminus S$
  - Απομένει λοιπόν η γεφύρωση της τομής $(S, V \setminus S)$ στην οποία αναγκαστικά τα , $'$ διαφέρουν. Έστω e' η ακμή που χρησιμοποιεί T'
  - $w(T') > w(T) \Rightarrow w(e') > w(e)$

Άσκηση 5 (2b): Απαραίτητες (και μη) ακμές

Θα δείξω: ακμή e απαραίτητη $\Leftrightarrow$ $\forall$ κύκλο C που την περιέχει, η e δεν είναι ακμή μέγιστου βάρους του C
Αρχικά ισχύουν ότι:

- Για κάθε κύκλο C, αν e μοναδική βαρύτερη ακμή $\Rightarrow$ κανένα MST δεν περιέχει την e
- Ξεκίνα από το G για όσο υπάρχει κύκλος, αφαίρεσε την βαρύτερη ακμή. Αν δεν υπάρχει κύκλος έχουμε ένα MST

Άσκηση 5 (2b): Απαραίτητες (και μη) ακμές

- Κατεύθυνση $\Rightarrow \exists\ e$ απαραίτητη:
  - Έστω T MST(G) και $e \in$ και $C_1, C_2, \ldots, C_n$ οι κύκλοι που περιέχουν την e. Αφού $e \in T \nexists C_i$: e μοναδική βαρύτερη
  - $T \backslash e$ ορίζει τομή $(S, V \backslash S)$. Έστω T' MST$(G \backslash e)$ το οποίο διαφέρει από το T μόνο στην ακμή $e' \in \delta(S, V \backslash S)$.
  - $T \cup e'$ και $T' \cup e$ θα περιέχουν τον ίδιο κύκλο C
  - Απο υπόθεση $w(T) < w(T') \Rightarrow w(e) < w(e')$
  - Παρόμοια μπορούμε να δείξουμε ότι $\forall$ κύκλο $C \in T \cap \delta(S, V \backslash S) \exists e' : w(e') > w(e)$

Άσκηση 5 (2b): Απαραίτητες (και μη) ακμές

- Κατεύθυνση $\Leftarrow \forall$ κύκλο C με $e \in C$ $e \neq \max\limits_{e' \in C}(w(e'))$
  - Έστω $\mathrm{MST}(G) : e \in T$
  - Έστω τ η τομή που ορίζει το $T \backslash e$
  - Κάθε $e' \in \delta(\tau) - T$, το $T \cup e'$ δημιουργεί ένα θεμελιώδη κύκλο C^θ
  - $\forall C^\theta, \forall e' \in C^\theta$ και $e' \notin T$ έχουμε
    - Από υπόθεση $w(e) \neq \max\limits_{e' \in C^\theta}(w(e'))$
    - $w(e) \leq w(e')$ ειδάλλως το θα είχε επιλέξει την e' για να διασχίσει την τομή τ
    - Κάποια από τις βαρύτερες ακμές του κύκλου δεν θα έχει επιλεγεί από το τ
  - Άρα χωρίς την e, η διάσχιση της τομής στοιχίζει κάτι παραπάνω!

Άσκηση 5 (3): Απαραίτητες (και μη) ακμές

Πρόβλημα: Είναι η ακμή e απαραίτητη; Απάντηση σε γραμμικό χρόνο.

- Αν e όχι μοναδική βαρύτερη ακμή σε έναν κύκλο $\Rightarrow$ η e είναι απαραίτητη
- BFS είναι γραμμικός και κάνει πλήρη εξερεύνηση του γράφου, άρα μπορούμε να εξετάσουμε βάρη ακμών σε κύκλους
- Αν αφαιρέσω όλες τις αυστηρά βαρύτερες ακμές και e όχι η μοναδική βαρύτερη για κάθε κύκλο $\Rightarrow$ η e δε θα ανήκει σε κύκλο

Άρα:

1. Αφαιρώ όλες τις ακμές που είναι αυστηρά πιο βαριές από την e. Αυτό μπορεί να γίνει σε γραμμικό χρόνο, απλά με ένα πέρασμα τις λίστας γειτνίασης $(|V| + |E|)$
2. Αν $e = (a,\ b)$, τρέχω BFS(a)
   - αν η εξερεύνηση με οδηγήσει στο a, η e ανήκει σε κύκλο για τον οποίο η e είναι η βαρύτερη ακμή $\rightarrow$ η e ΔΕΝ είναι απαραίτητη
   - Αν δεν μας οδηγήσει στο a τότε, είτε η e δεν άνηκε σε κανέναν κύκλο, είτε σε κάθε κύκλο που ανήκει δεν είναι η βαρύτερη

- Από ερώτημα β.2 έχουμε: e απαραίτητη ανν για κάθε κύκλο που περιέχει την e, αυτή δεν έχει το μέγιστο βάρος για τον κύκλο.
- Θα τρέξουμε τον Kruskal, μόνο που αντί να δοκιμάζουμε τις ακμές μία, μία θα εξετάζουμε όλες τις ακμές που έχουν ίδιο βάρος.

Άσκηση 5 (4): Απαραίτητες (και μη) ακμές

- Για κάθε τέτοιο σύνολο ακμών ίδιου βάρους
  - αρχικά δοκιμάζουμε τις ακμές μία μία
    - Όσες κάνουν κύκλο, δεν ανήκουν σε κανένα ΕΣΔ (αυστηρά πιο βαριές ακμές σε κάποιον κύκλο)
  - Όσες ακμές από την συστάδα απέμειναν τις δοκιμάζω όλες μαζί
    - Από αυτές κρατώ, ως απαραίτητες, όσες δεν κάνουν κύκλο