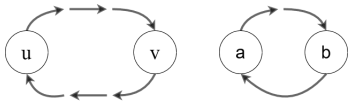
Λύσεις ασκήσεων φυλλαδίου 10 αλγορίθμων

Εαρινό 2020

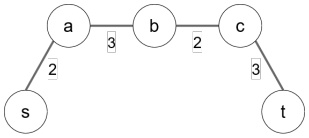
Άσκηση 1



Για κάθε κόμβο i εκτελώ Dijkstra σε $O(|V|^2)$ και βρίσκω τα μήκη των συντομότερων μονοπατιών από τον ι προς όλους τους άλλους. Συνολικά απαιτείται χρόνος: $O(|V|^3)$. Έστω ότι όλες αυτές τις αποστάσεις τις αποθηκεύω σε έναν πίνακα D, έτσι ώστε στη θέση $D(i,j)$ να βρίσκεται το μήκος του ελάχιστου μονοπατιού από τον i στον j. Για κάθε ζεύγος κόμβων u, v υπολογίζω το άθροισμα $D(u,v)+D(v,u)$ και επιστρέφω το μικρότερο από όλα αυτά τα αθροίσματα. Αν αυτό είναι ∞ τότε ο G είναι άκυκλος. Υπάρχουν $\binom{|V|}{2}$ ζεύγη κόμβων, άρα υπολογίζω $\binom{|V|}{2} = O(|V|^2)$ αθροίσματα σε χρόνο $O(1)$ το καθένα. Το ελάχιστο το βρίσκω σε γραμμικό χρόνο, διατρέχοντας μία φορά αυτά τα $O(|V|^2)$ αθροίσματα, άρα χρειάζεται $O(|V|^2)$ χρόνος. Συνολικά απαιτείται χρόνος: $O(|V|^3) + O(|V|^2) = O(|V|^3)$.

Άσκηση 2

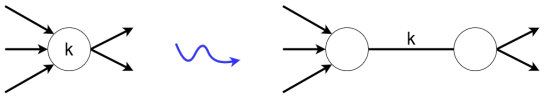
Παράδειγμα:
βάρη στις ακμές $\rightarrow$ κόστος $2+3+2+3=10$
βάρη σε ακμές και κόμβους $\rightarrow$ κόστος $10+4=14$



Έχουμε τις εξής εναλλακτικές:

1. Τροποποιούμε κατάλληλα τον αλγόριθμο Dijkstra έτσι ώστε στην αρχικοποίηση η ετικέτα κάθε κόμβου s να γίνεται $w(s)$ αντί για 0 και σε κάθε επανάληψη να ελέγχουμε αν ισχύει $L(u) > L(v) + w(u,v) + w(v)$ αντί για $L(u) > L(v) + w(v)$, και να ενημερώνουμε τα μονοπάτια ανάλογα.

2. Μετασχηματίζουμε κατάλληλα το γράφο αντικαθιστώντας κάθε κόμωο του με 2 άλλους μεταξύ τους με ακμή βάρους ίσου με το βάρος του αρχικού κόμβου και εκτελούμε Dijkstra στο γράφο που προκύπτει.



3. Μετασχηματίζουμε κατάλληλα το γράφο, σβήνοντας το βάρος κάθε κόμβου v και προσθέτοντας το στο βάρος κάθε ακμής που εισέρχεται στον v. Εκτελούμε Dijkstra στο γράφο που προκύπτει.

Άσκηση 3

$G = (V, A, W)$, $|V| = n$
Κάνουμε μία μικρή τροποποίηση στον αλγόριθμο Bellman- Ford για ακυκλικούς γράφους:
Ξεκινάμε αρχικοποιώντας τα κόστη των μονοπατιών σε - ∞ και ελέγχουμε:

$$V[i] + W[i, j] > V[j]$$

αντί για $V[j]$ κάθε φορά στον εσωτερικό κόμβο.

Αλγόριθμος (BF για DAGs)

$V = \{0, -\infty, -\infty, ...\}$

$p[1] = 1$

for $i = 1$ to n do {

for all j such that $(i, j) \in A$ and $V[i] + W[i, j] > V[j]$ do {

$V[j] = V[i] + W[i, j]$

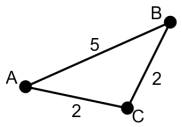
$p[j] = i$

}

}

Άσκηση 4

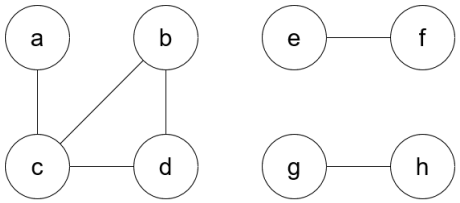
α΄ Όχι δε θα διατηρηθούν.
Αντιπαράδειγμα: Συντομότερο μονοπάτι από $A \to B : (AC, CB)$. Αν προσθέση τη $c = 2$ σε όλες τις ακμές συντομότερο μονοπάτι από $A \to B : (AB) \to 7$.



β΄ Ναι θα διατηρηθεί. Παραδείγματος χάριν, οι αλγόριθμοι Prim κα Kruskal θα εξετάσουν κόμβοθς και ακμές αντίστοιχα με την ίδια σειρά όπως πριν και θα εισάγουν τα ίδια στοιχεία, οπότε θα κατασκευάσουν το ίδιο δέντρο.

Άσκηση 5

Ο μη κατευθυνόμενος γράφος που παρτιστάνεται με τις λίστες $HEAD$ και $SUCC$ είναι ο εξής:



Τις συνεκτικές συνιστώσες του γράφου τις βρίσκω εκτελώντας τον εξής αλγόριθμο:
Αλγόριθμος BFS (Breath First Graph Traversal)

1. Θεωρούμε πίνακα $mark$ με $|V|$ στοιχεία.
2. Αρχικοποιούμε τον $mark$ σε $FALSE$.
3. $c \leftarrow 0$
4. for $i = 0$ to $|V|$ do{
   . if $mark[V_i] = FALSE$ then
   . $BFS(V_i)$
   . $c \leftarrow c + 1$
   . }
   }
5. Return c

Στο παράδειγμα μας, αρχικά, καλείται μία BFS με αρχή τον κόμβο a, η οποία θα ανακαλύψει τους κόμβους b, c, d άρα όλοι αυτοί μαζί αποτελούν μία συνεκτική συνιστώσα. Στη συνέχεια καλείται μία BFS με αρχή τον κόμβο e που ανακαλύπτει μόνο τον f (2η συνεκτική συνιστώσα). Τέλος, εκτελείται μία BFS με αρχή τον κόμβο g που ανακαλύπτει τον h (3η συνεκτική συνιστώσα). Δεν υπάρχουν άλλοι αμαρκάριστοι κόμβοι στο γράφο, οπότε ο αλγόριθμος έχει βρει όλες τις συνεκτικές συνιστώσες και τερματίζει.
Πολυπλοκότητα:

$$\Theta(n) + \sum_{i=1}^{c} \Theta(n_i + m_i) = \Theta(n) + \Theta(n+m) \approx \Theta(n+m),\ n = |V|\ \ m = |E|$$

Όπου c το πλήθοε των συνεκτικών συνιστωσών των γράφων και n_i, m_i το πλήθος κόμβων και ακμών αντίστοιχα της c-οστής συνεκτικής συνιστώσας αντίστοιχα.

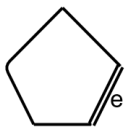
Άσκηση 6

α΄ Ξεκινάω με την AC επιλεγμένη και εκτελώ Prim/ Kruskal κατά τα γνωστά.

β΄ Σβήνω ακμές με βάρος < 3 (ή θέτω σε αυτές βάρος πολύ μεγάλο πχ ∞) και εκτελώ Prim/ Kruskal.

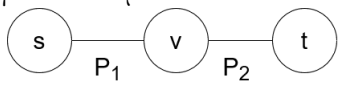
Άσκηση 7

Έστω ότι το ελάχιστο δέντρο επικάλυψης γράφου G περιέχει την ακμή e. Τότε, δεδομένου ότι η e ανήκει σε κύκλο, έστω τον c, μπορώ να προσθέσω μία από τις υπόλοιπες ακμές του c, έτσι ώστε να καλύψω όλους τους κόμβους του c. Τότε προκύπτει ελάχιστο δέντρο επικάλυψης με κόστος μικρότερο του προηγούμενου. ΑΤΟΠΟ! Αφού είχαμε θεωρήσει ελάχιστο δέντρο επικάλυψης. Άρα το ελάχιστο δέντρο επικάλυψης δεν περιέχει την ακμή e.



Άσκηση 8

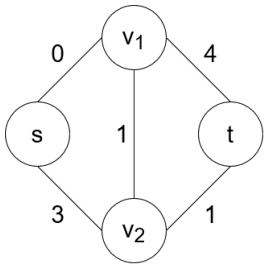
α΄ Δημιουργώ ένα νέο αλγόριθμο χρησιμοποιώντας την ιδέα του Dijkstra. Εναλλακτικά μπορώ να χρησιμοποιήσω ακριβώς τον αλγόριθμο Dijkstra, αρκεί να μετατρέψω κατάλληλα τα δεδομένα.



Η αξιοπιστία ενός μονοπατιού δίνεται από το γινόμενο τηα αξιοπιστίας των ακμών του $(p_1 \cdot p_2)$. Εμείς θέλουμε αθροιστοική συνάρτηση, επομένως θα πάρουμε τους λογαρίθμους της αξιοπιστίας των ακμών. Ο αλγόριθμος Dijkstra θέλει θετικά βάρη στις ακμές και υπολογίζει μονοπάτι ελαχίστου βάρους. Εμείς έχουμε αρνητικά βάρη και θέλουμε το μονοπάτι μέγιστου βάρους. Αν αλλάξουμε το πρόσημο όλων των λογαρίθμων τότε προκύπτει γράφος με θετικά βάρη στον οποίο αναζητούμε μονοπάτι ελαχίστου βάρους. Επομένως, χρησιμοποιούμε τον αλγόριθμο Dijkstra αφού αλλάξουμε το βάρος κάθε ακμής του γράφου από p σε $-logp$.

β′ Εφαρμόζοντας την παραπάνω μετατροπή τα βάρη γίνονται:

$$p'(s, v_1) = 0, \quad p'(s, v_2) = 3, \quad p'(v, t) = 4, \quad p'(v_1, v_2) = p'(v_2, t) = 1.$$



parent

s	v_1	v_2	t
s	X	X	X
s	s	s	X
s	s	v_1	v_1
s	s	v_1	v_2
s	s	v_1	v_2

distance

s	v_1	v_2	t
0	∞	∞	∞
0	0	3	∞
0	0	1	4
0	0	1	2
0	0	1	2

fixed

s	v_1	v_2	t
X	X	X	X
√	X	X	X
√	√	X	X
√	√	√	X
√	√	√	√