# <u>ΑΛΓΟΡΙΘΜΟΙ ΚΑΙ ΠΟΛΥΠΛΟΚΟΤΗΤΑ</u>, ΚΑΝΟΝΙΚΗ 2021

## <u>Μέρος Α</u>

3. Περιγράψτε συνοπτικά τον καλύτερο αλγόριθμο που μπορείτε να βρείτε για το παρακάτω πρόβλημα. Να αναφέρετε και να αιτιολογήσετε πολύ σύντομα την υπολογιστική πολυπλοκότητα του αλγόριθμου που περιγράψατε:

   Έστω δύο ταξινομημένοι (σε αύξουσα σειρά) πίνακες A[1 … n] και B[1 … m], με n και m φυσικούς αριθμούς, και δύο ακέραιοι αριθμοί L και M, όπου L < M. Ένα ζευγάρι θέσεων ( i , j ), όπου η θέση i, 1 ≤ i ≤ n, βρίσκεται στον πίνακα A και η θέση j, 1 ≤ j ≤ m, βρίσκεται στον πίνακα B, θεωρείται έγκυρο αν L ≤ B[ j ] – A[ i ] ≤ M. Θέλουμε να υπολογίσουμε το συνολικό πλήθος των έγκυρων ζευγαριών θέσεων ( i , j ) στους πίνακες A και B.

4. Η ταξινόμηση με εισαγωγή, σε έναν μη ταξινομημένο πίνακα n στοιχείων, όταν η θέση στην οποία πρέπει να εισαχθεί το επόμενο στοιχείο υπολογίζεται με δυαδική αναζήτηση, έχει χρόνο εκτέλεσης χειρότερης περίπτωσης:

   - Θ(n (log n)^2)
   - Θ(n^2)
   - Θ(n)
   - Θ(n log n)
   - Κανένα από τα άλλα
   - Δεν απαντώ

5. Θεωρούμε το δέντρο αναδρομής που περιγράφει την εκτέλεση του αλγόριθμου quicksort, όταν αυτός εφαρμόζεται σε έναν μη ταξινομημένο πίνακα n στοιχείων. Ο χρόνος εκτέλεσης που δαπανάται σε κάθε επίπεδο του δέντρου αναδρομής, με εξαίρεση ίσως το τελευταίο, είναι της ίδιας τάξης μεγέθους.

   - Δεν απαντώ.
   - Λάθος
   - Σωστό

6. Ο χρόνος εκτέλεσης του αλγόριθμου Radix Sort, όταν αυτός εφαρμόζεται σε μη ταξινομημένο πίνακα n φυσικών αριθμών που δεν ξεπερνούν το M, είναι:

- Θ(n+M)
- Κανένα από τα άλλα
- Θ(n)
- Δεν απαντώ
- O( n log M )

7. Σε ένα ATM, το οποίο έχει ελεύθερη μνήμη μόλις λίγα KB (λίγες χιλιάδες bytes), θέλετε να ταξινομήσετε μερικά εκατομμύρια συναλλαγές που αφορούν σε αναλήψεις χρημάτων, σε φθίνουσα σειρά ως προς το ποσό ανάληψης που αφορούσε η καθεμία (η μνήμη των λίγων KB παραμένει ελεύθερη μετά την αποθήκευση των συναλλαγών που θέλετε να ταξινομήσετε). Μπορείτε να θεωρήσετε ότι τα ποσά ανάληψης ποικίλουν σημαντικά. Ποιον από τους παρακάτω αλγόριθμους ταξινόμησης θα προτιμήσετε για την ταξινόμηση των συναλλαγών:

- Counting sort
- Δεν απαντώ
- Radix sort
- Ταξινόμηση με εισαγωγή (insertion sort)
- Ταξινόμηση σωρού (heapsort)
- Ταξινόμηση με συγχώνευση (merge sort)

8. Έστω ένα δέντρο T με ρίζα και n κόμβους, στο οποίο θέλουμε να απαντάμε ερωτήματα Lowest Common Ancestor (LCA, ποιος είναι ο κοντινότερος κοινός πρόγονος δύο δεδομένων κόμβων u και v). Είναι δυνατόν να απαντάμε τέτοια ερωτήματα σε χρόνο $O(1)$, αφού έχουμε δαπανήσει χρόνο προεπεξεργασίας $O(n)$.

Σωστό

Λάθος

Δεν απαντώ

9. Σε έναν σωρό μεγίστου με n διαφορετικά στοιχεία, το ελάχιστο στοιχείο μπορεί να υπολογισθεί σε χρόνο $O(\log n)$.

Σωστό

Λάθος

Δεν απαντώ

10. Δίνεται η παρακάτω ακολουθία θετικών φυσικών αριθμών: (697, 478, 468, 657, 620, 470, 891, 821, 159, <Τρία Τελευταία Ψηφία του ΑΜ σας>). Να δώσετε την ακολουθία στην οποία καταλήγει ο αλγόριθμος RadixSort, αμέσως μετά την ολοκλήρωση της 2ης φάσης εκτέλεσής του.

11. Η αναδρομική σχέση $T(n) = T(n/3)+T(2*n/3)+\Theta(n \log(n))$, με $T(1) = \Theta(1)$, έχει λύση:

T(n) = Θ(n)

T(n) = Θ(n log(n))

Δεν απαντώ

Κανένα από τα άλλα

T(n) = Θ( n^(log_3(2)) )

T(n) = Θ(n (log(n))^2)

12. Η αναδρομική σχέση T(n) = (log n)^2 + T(n^(1/2)), με T(1) = Θ(1), έχει λύση:

- T(n) = Θ(((log n)^2) *(loglog n) )
- T(n) = Θ((log n)*(loglog n))
- T(n) = Θ((log n)*n^(1/2))
- Κανένα από τα άλλα
- Δεν απαντώ
- T(n) = Θ( (log n)^2 )

13. Σε μια συλλογή n στοιχείων, μπορούμε να απαντήσουμε m >= n ερωτήματα find και να εκτελέσουμε n-1 unions σε χρόνο:

- o(m+n)
- o(m log n)
- Δεν απαντώ
- O(m^(3/4) + n)
- Κανένα από τα άλλα.

14. Θεωρούμε τον αναδρομικό αλγόριθμο funAlgo που περιγράφεται παρακάτω και εφαρμόζεται σε έναν πίνακα A. Ειδικότερα, η κλήση FunAlgo(A, low, up) αφορά τον υποπίνακα A[low..up], δηλ. τα στοιχεία στις θέσεις από low μέχρι και up του πίνακα A.

```
funAlgo(array A[ ], low, up)
    if (low >= up) return;
    divide(A, low, up);
    mid = (low+up)/2;
    funAlgo(A, low, mid);
    funAlgo(A, mid+1, up);
    conquer(A, low, up);
```

Η διαδικασία divide έχει χρονική πολυπλοκότητα Θ(n), αν εφαρμοστεί σε υποπίνακα του Α που περιλαμβάνει n στοιχεία. Η διαδικασία conquer έχει χρονική πολυπλοκότητα Θ(n log n), αν εφαρμοστεί σε υποπίνακα του Α που περιλαμβάνει n στοιχεία. Ποια (μία μόνο) από τις παρακάτω προτάσεις για την χρονική πολυπλοκότητα της κλήσης funAlgo(A, 1, n) είναι αληθής (θεωρούμε ότι ο πίνακας Α έχει τουλάχιστον n στοιχεία);

Η χρονική πολυπλοκότητα T(n) της κλήσης funAlgo(A, 1, n) περιγράφεται από την αναδρομική σχέση T(n) = 2 T(n / 2) + Θ(n) + Θ(n log n), με T(1) = Θ(1), και είναι T(n) = Θ(n^2 log n).

Δεν απαντώ

Η χρονική πολυπλοκότητα T(n) της κλήσης funAlgo(A, 1, n) περιγράφεται από την αναδρομική σχέση T(n) = 2 T(n / 2) + Θ(n log n), με T(1) = Θ(1), και είναι T(n) = Θ(n^(3/2) log n).

Η χρονική πολυπλοκότητα T(n) της κλήσης funAlgo(A, 1, n) περιγράφεται από την αναδρομική σχέση T(n) = 2 T(n / 2) + Θ(n) + Θ(n log n), με T(1) = Θ(1), και είναι T(n) = Θ(n log n).

Η χρονική πολυπλοκότητα T(n) της κλήσης funAlgo(A, 1, n) περιγράφεται από την αναδρομική σχέση T(n) = 2 T(n / 2) + Θ(n log n), με T(1) = Θ(1), και είναι T(n) = Θ(n (log n)^2).

Κανένα από τα άλλα

## Μέρος Β

3

(2 μονάδες) Να διατυπώσετε ένα αντι-παράδειγμα, όπου ο "φυσιολογικός" άπληστος αλγόριθμος (που περιγράψαμε στο μάθημα) αποτυγχάνει να υπολογίσει τη βέλτιστη λύση για το παρακάτω πρόβλημα:

Έστω σύνολο X = { x1, ..., xk } θετικών φυσικών αριθμών, που περιλαμβάνει το 1, και έστω φυσικός αριθμός B. Θέλουμε να διατυπώσουμε το B ως άθροισμα στοιχείων του X με το ελάχιστο πλήθος προσθετέων (το ίδιο στοιχείο του X μπορεί να χρησιμοποιηθεί όσες φορές θέλουμε στο άθροισμα).

4

Τι ισχύει για τις παρακάτω προτάσεις:

1. Σε έναν αλγόριθμο δυναμικού προγραμματισμού, ο υπολογισμός της λύσης με bottom-up τρόπο μπορεί να είναι ασυμπτωτικά ταχύτερος από τον υπολογισμό της λύσης με top-down τρόπο και απομνημόνευση (memoization).

2. Μπορούμε να χρησιμοποιήσουμε δυαδική αναζήτηση για να ανάγουμε πολυωνυμικά ένα πρόβλημα συνδυαστικής βελτιστοποίησης (π.χ., υπολογισμός περιοδείας ελάχιστου μήκους σε δεδομένο στιγμιότυπο του Προβλήματος του Πλανόδιου Πωλητή) στο αντίστοιχο πρόβλημα απόφασης (π.χ., δεδομένου ενός στιγμιότυπου του Προβλήματος του Πλανόδιου Πωλητή και φυσικού B, υπάρχει περιοδεία μήκους το πολύ B;).

Η 1η είναι σωστή και η 2η είναι λάθος.

Δεν απαντώ

Η 1η είναι λάθος και η 2η είναι λάθος.

Η 1η είναι σωστή και η 2η είναι σωστή.

Η 1η είναι λάθος και η 2η είναι σωστή.

5

Ένα κατευθυνόμενο ακυκλικό γράφημα G έχει μοναδική τοπολογική διάταξη αν και μόνο αν το G έχει μονοπάτι Hamilton, δηλ. ένα απλό κατευθυνόμενο μονοπάτι που περνάει από όλες τις κορυφές.

Λάθος

Σωστό

Δεν απαντώ

6

Τι ισχύει για τις παρακάτω προτάσεις:

1. Ο χρόνος εκτέλεσης ενός αλγόριθμου δυναμικού προγραμματισμού είναι πάντα Θ(Κ), όπου Κ είναι το πλήθος των υποπροβλημάτων που προκύπτουν από την αναδρομική σχέση στην οποία βασίζεται ο αλγόριθμος.

2. Κάθε πρόβλημα μεγιστοποίησης κέρδους που λύνεται με δυναμικό προγραμματισμό μπορεί να αναχθεί σε ένα πρόβλημα εύρεσης μακρύτερου μονοπατιού σε ένα κατευθυνόμενο ακυκλικό γράφημα (DAG, directed acyclic graph).

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι λάθος.

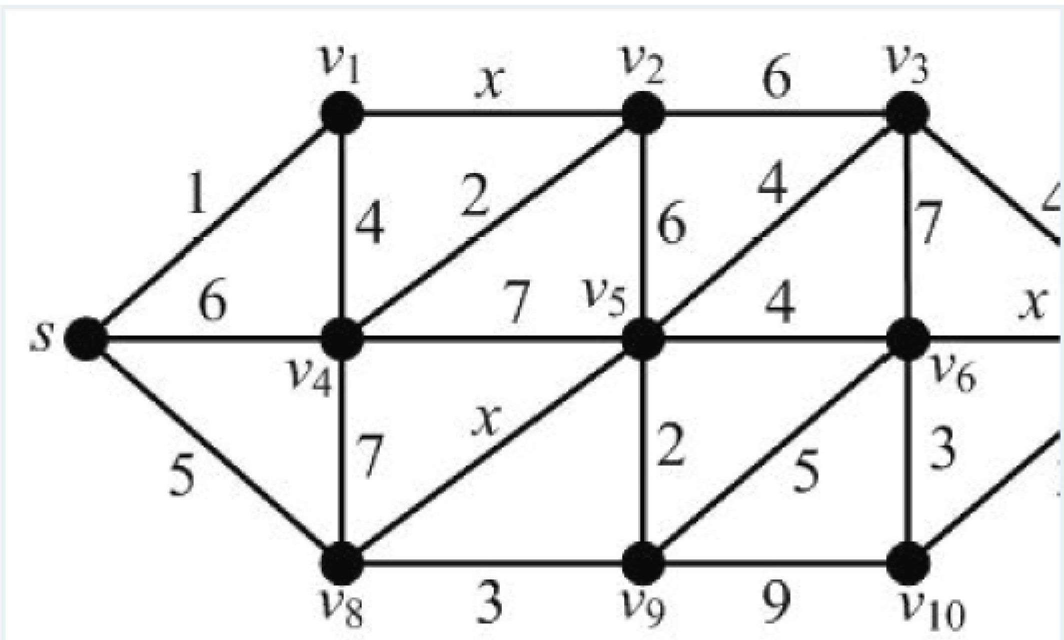
Δεν απαντώ

Η 1η είναι σωστή και η 2η είναι σωστή.

Η 1η είναι λάθος και η 2η είναι λάθος.

7

(4 μονάδες) Στο παρακάτω γράφημα, να υπολογίσετε ένα Ελάχιστο Συνδετικό Δέντρο χρησιμοποιώντας τον αλγόριθμο Prim με αρχική κορυφή την s. Για αυτό το σκοπό, να θέσετε όπου x το τελευταίο ψηφίο του ΑΜ σας. Ποιο είναι το βάρος του Ελάχιστου Συνδετικού Δέντρου; Να αναφέρετε ποιες ακμές και με ποια σειρά προστίθενται στο Ελάχιστο Συνδετικό Δέντρο.



8

Τι ισχύει για τις παρακάτω προτάσεις:

1. Σε μη κατευθυνόμενο γράφημα με $n$ κορυφές και $m = \Theta(n^{(3/2)})$ ακμές που αναπαρίσταται με πίνακα γειτνίασης, το DFS χρειάζεται χρόνο $O(n^{(3/2)})$.

2. Μπορούμε να αποφανθούμε σε γραμμικό χρόνο, αν σε ένα μη κατευθυνόμενο γράφημα με $n$ κορυφές και $m > n$ ακμές, όλοι οι κύκλοι έχουν άρτιο μήκος.

Η 1η είναι λάθος και η 2η είναι λάθος.

Δεν απαντώ

Η 1η είναι σωστή και η 2η είναι λάθος.

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι σωστή.

9

Τι ισχύει για τις παρακάτω προτάσεις:

1. Σε ένα μη κατευθυνόμενο συνεκτικό γράφημα με βάρη στις ακμές, κάθε ακμή ελάχιστου βάρους ανήκει σε κάποιο Ελάχιστο Συνδετικό Δέντρο.

2. Σε ένα μη κατευθυνόμενο συνεκτικό γράφημα με $n$ κορυφές και $n+30$ ακμές, μπορούμε να υπολογίσουμε ένα Ελάχιστο Συνδετικό Δέντρο σε γραμμικό χρόνο.

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι σωστή.

Δεν απαντώ

Η 1η είναι λάθος και η 2η είναι λάθος.

Η 1η είναι σωστή και η 2η είναι λάθος.

10

Η βέλτιστη λύση για τη Διακριτή εκδοχή του προβλήματος του Σακιδίου περιέχει πάντα το αντικείμενο με τον μέγιστο λόγο αξίας προς μέγεθος.

Σωστό

Λάθος

Δεν απαντώ

11

(4 μονάδες) Περιγράψτε συνοπτικά τον καλύτερο αλγόριθμο που γνωρίζετε για το παρακάτω πρόβλημα. Να αναφέρετε και να αιτιολογήσετε πολύ σύντομα την υπολογιστική πολυπλοκότητα του αλγόριθμου που περιγράψατε:

Θεωρούμε μια ακολουθία Α = (α1, ..., αn) αποτελούμενη από n φυσικούς αριθμούς. Θέλουμε να υπολογίσουμε μια μέγιστη υπακολουθία της Α που είναι σχεδόν αύξουσα. Μια υποακολουθία Β = (β1, ..., βk) της Α είναι σχεδόν αύξουσα αν είναι η Β είναι γνήσια αύξουσα ή υπάρχει στη Β ένα μοναδικό ζευγάρι διαδοχικών στοιχείων β(i) και β(i+1) για το οποίο ισχύει ότι β(i) >= β(i+1) (για τα υπόλοια έχουμε ότι β1 < ... < β(i) και β(i+1) < ... < β(k) ).

12

Τι ισχύει για τις παρακάτω προτάσεις που σχετίζονται με το πρόβλημα της επιλογής ενός μέγιστου πλήθους μη επιλυπτόμενων διαστημάτων:

1. Ο άπληστος αλγόριθμος που επιλέγει το διαθέσιμο διάστημα (δηλ. το διάστημα που δεν επικαλύπτεται με τα διαστήματα που έχουμε ήδη επιλέξει) με τη μικρότερη διάρκεια μεταξύ όλων των διαθέσιμων διαστημάτων υπολογίζει πάντα τη βέλτιστη λύση.

2. Αν όλα τα διαστήματα έχουν το ίδιο μήκος, ο άπληστος αλγόριθμος που επιλέγει το διαθέσιμο διάστημα (δηλ. το διάστημα που δεν επικαλύπτεται με τα διαστήματα που έχουμε ήδη επιλέξει) που ξεκινά πρώτο υπολογίζει πάντα τη βέλτιστη λύση.

Η 1η είναι σωστή και η 2η είναι σωστή.

Δεν απαντώ

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι λάθος.

Η 1η είναι λάθος και η 2η είναι λάθος.

13

Ο χρόνος εκτέλεσης του αλγόριθμου δυναμικού προγραμματισμού για τη Διακριτή εκδοχή του προβλήματος του Σακιδίου είναι πολυωνυμικός.

Λάθος

Δεν απαντώ

Σωστό

14

Τι ισχύει για τις παρακάτω προτάσεις:

1. Μπορούμε να υπολογίσουμε όλες τις γέφυρες και τα σημεία κοπής ενός μη κατευθυνόμενου συνεκτικού γραφήματος σε γραμμικό χρόνο με Αναζήτηση κατά Πλάτος (BFS).

2. Μπορούμε να υπολογίσουμε μια τοπολογική διάταξη ενός κατευθυνόμενου ακυκλικού γραφήματος σε γραμμικό χρόνο με Αναζήτηση κατά Βάθος (DFS).

Η 1η είναι σωστή και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι λάθος.

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι λάθος και η 2η είναι λάθος.

Δεν απαντώ

## <u>Μέρος Γ</u>

1

Δίνεται μη κατευθυνόμενο συνεκτικό γράφημα G με θετικά βάρη στις ακμές. Υπάρχει αλγόριθμος γραμμικού χρόνου για τον υπολογισμό ενός Συνδετικού Δέντρου του G στο οποίο η βαρύτερη ακμή έχει το ελάχιστο δυνατό βάρος.

Σωστό

Λάθος

Δεν απαντώ

2

Τι ισχύει για τις παρακάτω προτάσεις:

1. Αν στον αλγόριθμο Dijkstra, επιλέγουμε σε κάθε επανάληψη τη διαθέσιμη κορυφή με τη μέγιστη πεπερασμένη ετικέτα, τότε υπολογίζουμε τα μονοπάτια μέγιστου μήκους από την αρχική κορυφή s προς κάθε άλλη κορυφή.

2. Έστω κατευθυνόμενο γράφημα με θετικά μήκη στις ακμές, και έστω p ένα συντομότερο μονοπάτι από μια κορυφή s σε μια κορυφή t. Το p παραμένει ένα συντομότερο s – t μονοπάτι αν αντικαταστήσουμε το μήκος κάθε ακμής με την τετραγωνική του ρίζα.

Η 1η είναι λάθος και η 2η είναι λάθος.

Η 1η είναι σωστή και η 2η είναι σωστή.

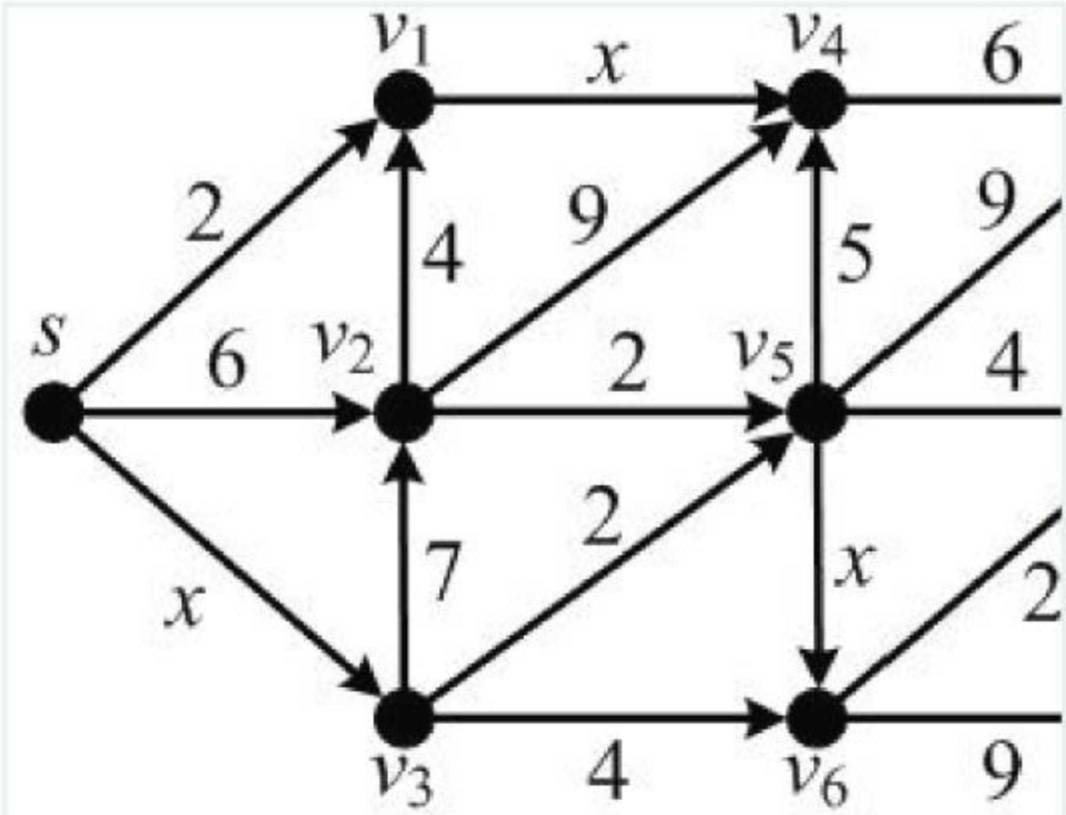
Δεν απαντώ

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι λάθος.

3

Στο παρακάτω γράφημα, να υπολογίσετε τις αποστάσεις και τα συντομότερα μονοπάτια όλων των κορυφών από την s χρησιμοποιώντας τον αλγόριθμο του Dijkstra. Για αυτό το σκοπό, να θέσετε όπου x το τελευταίο ψηφίο του ΑΜ σας. Να αναφέρετε ποιες κορυφές, με ποια σειρά, και με ποια απόσταση από την s προστίθενται στο Δέντρο Συντομότερων Μονοπατιών.



$v_1$
$x$
$v_4$
6
2
4
9
5
9
$s$
6
$v_2$
2
$v_5$
4
2
7
$x$
$x$
2
$v_3$
4
$v_6$
9

4

Τι ισχύει για τις παρακάτω προτάσεις:

1. Για ένα δίκτυο ροής G με n κορυφές, m >= n ακμές, και ακέραιες χωρητικότητες στις ακμές του, ο αλγόριθμος Ford–Fulkerson έχει χρόνο εκτέλεσης χειρότερης περίπτωσης O(m |f|), όπου |f| είναι το μέγεθος της μέγιστης ροής.

2. Αν σε ένα δίκτυο ροής, αυξήσουμε τη χωρητικότητα όλων των ακμών της ελάχιστης τομής κατά 1, τότε η μέγιστη ροή θα αυξηθεί αναγκαστικά τουλάχιστον κατά 1.

Η 1η είναι σωστή και η 2η είναι λάθος.

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι σωστή.

Δεν απαντώ

Η 1η είναι λάθος και η 2η είναι λάθος.

5

Τι ισχύει για τις παρακάτω προτάσεις:

1. Αν ένα πρόβλημα Π ανάγεται πολυωνυμικά σε ένα γνωστό NP-δύσκολο πρόβλημα, τότε το Π είναι αναγκαστικά NP-δύσκολο.

2. Αν η κλάση P ταυτίζεται με την κλάση NP, τότε όλα τα προβλήματα στην κλάση NP είναι NP-πλήρη.

Η 1η είναι λάθος και η 2η είναι λάθος.

Η 1η είναι σωστή και η 2η είναι λάθος.

Δεν απαντώ

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι σωστή.

6

Έστω κατευθυνόμενο γράφημα G με n κορυφές, m = n^(3/2) ακμές και (ενδεχομένως αρνητικά) βάρη στις ακμές του. Ο αλγόριθμος του Johnson στο G, για τον υπολογισμό των συντομότερων μονοπατιών μεταξύ όλων των ζευγών κορυφών, είναι ασυμπτωτικά ταχύτερος από τον αλγόριθμο Floyd-Warshall.

Δεν απαντώ

Σωστό

Λάθος

8

Τι ισχύει για τις παρακάτω προτάσεις:

1. Το παρακάτω πρόβλημα ανήκει στην κλάση NP: δεδομένου ενός σύνθετου αριθμού n, που γράφεται ως γινόμενο δύο πρώτων αριθμών, να υπολογίσουμε τους δύο πρώτους παράγοντες του n.

2. Κάθε πρόβλημα στο NP μπορεί να λυθεί σε εκθετικό χρόνο.

Η 1η είναι σωστή και η 2η είναι λάθος.

Δεν απαντώ

Η 1η είναι λάθος και η 2η είναι λάθος.

Η 1η είναι σωστή και η 2η είναι σωστή.

Η 1η είναι λάθος και η 2η είναι σωστή.

9

Δίνεται κατευθυνόμενο ισχυρά συνεκτικό γράφημα G με θετικά μήκη στις ακμές και θετικός ακέραιος B. Υπάρχει αλγόριθμος γραμμικού χρόνου που υπολογίζει μονοπάτι p μεταξύ δύο συγκεκριμένων κορυφών, s και t, του G τέτοιο ώστε κάθε ακμή του p να έχει μήκος το πολύ B (ή αποφαίνεται ότι δεν υπάρχει τέτοιο μονοπάτι).

Δεν απαντώ

Λάθος

Σωστό

10

Θεωρούμε μια εφαρμογή του αλγόριθμου συντομότερων μονοπατιών του Dijkstra σε κατευθυνόμενο γράφημα G με n κορυφές, m ακμές και μη αρνητικά βάρη στις ακμές του. Θεωρούμε την περίπτωση που το G είναι ισχυρά συνεκτικό. Ποια (μόνο μία) από τις παρακάτω προτάσεις είναι αληθής:

Μπορεί να υπάρξει ακμή e = (u, v) που εξετάζεται από τον αλγόριθμο Θ(n) φορές, στη χειρότερη περίπτωση, για πιθανή αναπροσαρμογή της εκτίμησης απόστασης της v.

Ο αλγόριθμος εξετάζει κάθε ακμή e = (u, v) Θ(n) φορές, στη χειρότερη περίπτωση, για πιθανή αναπροσαρμογή της εκτίμησης απόστασης της v.

Ο αλγόριθμος εξετάζει κάθε ακμή e = (u, v) μία φορά για πιθανή αναπροσαρμογή της εκτίμησης απόστασης της v.

Κανένα από τα άλλα

Δεν απαντώ

Μπορεί να υπάρξει ακμή e = (u, v) που δεν εξετάζεται καθόλου από τον αλγόριθμο για πιθανή αναπροσαρμογή της εκτίμησης απόστασης της v.

11

Έστω κατευθυνόμενο γράφημα G με (ενδεχομένως αρνητικά) βάρη στις ακμές του. Τι ισχύει για τις παρακάτω προτάσεις:

1. Αν το G δεν περιέχει κύκλους μηδενικού ή θετικού μήκους, τότε ο αλγόριθμος Bellman-Ford μπορεί να υπολογίσει ένα μακρύτερο μονοπάτι (δηλ. ένα μονοπάτι μέγιστου συνολικού βάρους) από μια συγκεκριμένη κορυφή s προς μια συγκεκριμένη κορυφή t.

2. Αν το G περιέχει έναν κύκλο αρνητικού μήκους, ο αλγόριθμος Bellman-Ford υπολογίζει μια συντομότερη απλή διαδρομή από μια συγκεκριμένη κορυφή s προς μια συγκεκριμένη κορυφή t.

Η 1η είναι σωστή και η 2η είναι λάθος.

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι λάθος και η 2η είναι λάθος.

Δεν απαντώ

Η 1η είναι σωστή και η 2η είναι σωστή.

13

Περιγράψτε συνοπτικά τον καλύτερο αλγόριθμο που γνωρίζετε για το παρακάτω πρόβλημα. Να αναφέρετε και να αιτιολογήσετε πολύ σύντομα την υπολογιστική πολυπλοκότητα του αλγόριθμου που περιγράψατε:

Θεωρούμε ένα κατευθυνόμενο γράφημα G με n κορυφές, m ακμές, και θετικό μήκος w(e) σε κάθε ακμή του e. Κάποιες ακμές G είναι χρωματισμένες κόκκινες, ενώ οι υπόλοιπες ακμές του G είναι χρωματισμένες πράσινες. Έστω δύο κορυφές s, t του G. Για δεδομένο k, θέλουμε να υπολογίζουμε το συντομότερο s - t μονοπάτι που περιέχει το πολύ k κόκκινες ακμές.

14

Έστω ένα συνεκτικό μη κατευθυνόμενο γράφημα G με θετικά βάρη στις ακμές του. Τι ισχύει για τις παρακάτω προτάσεις:

1. Τα συντόμοτερα μονοπάτια του G από μια αρχική κορυφή s προς όλες τις άλλες δεν μεταβάλλονται αν προσθέσουμε σε όλες τις ακμές του G τον ίδιο θετικό αριθμό k.

2. Το Ελάχιστο Συνδετικό Δέντρο του G δεν μεταβάλλεται αν προσθέσουμε σε όλες τις ακμές του G τον ίδιο θετικό αριθμό k.

Η 1η είναι σωστή και η 2η είναι σωστή.

Η 1η είναι λάθος και η 2η είναι λάθος.

Η 1η είναι λάθος και η 2η είναι σωστή.

Η 1η είναι σωστή και η 2η είναι λάθος.

Δεν απαντώ