Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

Αλγόριθμοι και Πολυπλοκότητα

Διδάσκοντες: Σ. Ζάχος, Δ. Φωτάκης

5η Σειρά Γραπτών Ασκήσεων - Ημ/νία Παράδοσης 12/2/2010

Άσκηση 1: Βελτίωση Οδικού Δικτύου

Έστω οδικό δίκτυο $G(V, E, \ell)$ που συνδέει ένα σύνολο πόλεων V. Θεωρούμε ότι το δίκτυο είναι κατευθυνόμενο, και ότι κάθε δρόμος $(u, v) \in E$ έχει (μη αρνητικό) μήκος $\ell(u, v)$. Πρόκειται να κατασκευαστεί ένας νέος δρόμος, και υπάρχει λίστα E' με προτάσεις για ζεύγη πόλεων τα οποία μπορεί να συνδέσει. Κάθε πρόταση $(u, v) \in E'$ συνοδεύεται από αντίστοιχο μήκος $\ell'(u, v)$. Το ζητούμενο είναι να επιλέξουμε την πρόταση που επιτυγχάνει την μέγιστη μείωση της απόστασης μεταξύ δύο δεδομένων πόλεων $s, t \in V$. Να διατυπώσετε έναν όσο το δυνατόν πιο αποδοτικό αλγόριθμο για αυτό το πρόβλημα. Να αιτιολογήσετε την ορθότητα και την πολυπλοκότητα του αλγορίθμου σας. *Σημείωση:* Πρόκειται για την Άσκηση **4.10** του DPV.

Άσκηση 2: Μείωση Μεταφορικής Ικανότητας

Θεωρούμε δίκτυο ροής $G(V, E)$ με αρχική κορυφή $s \in V$, καταληκτική κορυφή $t \in V$, και μοναδιαίες χωρητικότητες στις ακμές, και φυσικό αριθμό k. Το ζητούμενο είναι να βρούμε k ακμές η διαγραφή των οποίων προκαλεί την μεγαλύτερη δυνατή μείωση στην μέγιστη ροή μεταξύ των κορυφών s και t. Να διατυπώσετε έναν αποδοτικό αλγόριθμο για αυτό το πρόβλημα. Να αιτιολογήσετε την ορθότητα του αλγορίθμου σας, και να διερευνήσετε αν η προσέγγιση που ακολουθήσατε γενικεύεται για ακμές διαφορετικής χωρητικότητας. *Σημείωση:* Πρόκειται για την Άσκηση **7.12** του KT.

Άσκηση 3: Προγραμματισμός Εφημεριών

Ένα νοσοκομείο που απασχολεί k γιατρούς προγραμματίζει τις εφημερίες τους για τις επόμενες n ημέρες. Κάθε γιατρός j, $1 \le j \le k$, έχει δηλώσει ένα σύνολο ημερών L_j στις οποίες είναι διαθέσιμος για εφημερία. Επιπλέον, για κάθε ημέρα i, $1 \le i \le n$, είναι γνωστός ο ακριβής αριθμός p_i των γιατρών που εφημερεύουν. Πρέπει λοιπόν να υπολογισθεί ένα σύνολο ημερών εφημερίας $L'_j \subseteq L_j$ για κάθε γιατρό j, ώστε κάθε ημέρα i να περιλαμβάνεται στα σύνολα ημερών εφημερίας ακριβώς p_i γιατρών, ή να διαπιστωθεί ότι κάτι τέτοιο δεν είναι εφικτό. Να διατυπώσετε έναν αποδοτικό αλγόριθμο για αυτό το πρόβλημα. Να αιτιολογήσετε την ορθότητα και την πολυπλοκότητα του αλγορίθμου σας. *Σημείωση:* Πρόκειται για την Άσκηση **7.19** του KT.

Άσκηση 4: Αναγωγές και NP-Πληρότητα

Να δείξετε ότι τα παρακάτω προβλήματα είναι **NP**-Πλήρη:

Συνδετικό Δέντρο με Ελάχιστο Αριθμό Φύλλων (Min Leaf Spanning Tree)

Είσοδος: Μη κατευθυνόμενο γράφημα $G(V, E)$ και φυσικός αριθμός k, $2 \le k < |V|$.

Ερώτηση: Έχει το G συνδετικό δέντρο με k ή λιγότερα φύλλα (δηλ. κορυφές βαθμού 1);

Κυρίαρχο Σύνολο (Dominating Set)

Είσοδος: Μη κατευθυνόμενο γράφημα $G(V, E)$ και φυσικός αριθμός k, $1 \leq k \leq |V|$.

Ερώτηση: Έχει το G κυρίαρχο σύνολο $D \subseteq V$ με $|D| \leq k$; Ένα σύνολο κορυφών $D \subseteq V$ αποτελεί κυρίαρχο σύνολο για ένα μη κατευθυνόμενο γράφημα $G(V, E)$ αν κάθε κορυφή $v \in V$ είτε ανήκει στο D είτε συνδέεται με κάποια κορυφή του D.

Δέντρο Steiner (Steiner Tree)

Είσοδος: Πλήρες μη κατευθυνόμενο γράφημα $G(V, E)$ με μήκη $\ell : E \mapsto \mathbb{N}$ στις ακμές, υποσύνολο κορυφών $C \subset V$, και φυσικός αριθμός B. Τα μήκη των ακμών ικανοποιούν την τριγωνική ανισότητα, δηλ. για κάθε $x, y, z \in V$, $\ell(x, y) \leq \ell(x, z) + \ell(z, y)$.

Ερώτηση: Υπάρχει συνεκτικό υπογράφημα του G που περιλαμβάνει όλες τις κορυφές του C (και ενδεχομένως κάποιες κορυφές του $V \setminus C$) με συνολικό μήκος που δεν ξεπερνά το B;

Σημείωση: Μπορούμε να θεωρήσουμε ότι το ζητούμενο υπογράφημα είναι δέντρο. Ένα τέτοιο δέντρο ονομάζεται *δέντρο Steiner* για το C. Το πρόβλημα είναι διαφορετικό από τον υπολογισμό ενός Ελάχιστου Συνδετικού Δέντρου, αφού το βέλτιστο δέντρο Steiner δεν περιλαμβάνει κατ' ανάγκη όλες τις κορυφές του V, αλλά μπορεί να εκμεταλλευθεί την ύπαρξη κάποιων κορυφών του $V \setminus C$ για να επιτύχει μικρότερο συνολικό μήκος σε σχέση με το συνολικό μήκος ενός Ελάχιστου Συνδετικού Δέντρου στο επαγόμενο υπογράφημα που ορίζεται από το C.

Χωροθέτηση Υπηρεσιών (Facility Location)

Είσοδος: Πλήρες μη κατευθυνόμενο γράφημα $G(V, E)$ με αποστάσεις $\ell : E \mapsto \mathbb{N}$ στις ακμές και κόστη $f : V \mapsto \mathbb{N}$ στις κορυφές, και φυσικός αριθμός B. Οι αποστάσεις ικανοποιούν την τριγωνική ανισότητα, δηλ. για κάθε $x, y, z \in V$, $\ell(x, y) \leq \ell(x, z) + \ell(z, y)$.

Ερώτηση: Υπάρχει σύνολο κορυφών $C \subseteq V$ για το οποίο το συνολικό κόστος των κορυφών του C και η συνολική απόσταση κάθε κορυφής του $V \setminus C$ από την πλησιέστερη κορυφή στο C δεν ξεπερνά το B; Πρέπει δηλαδή να διαπιστώσουμε αν υπάρχει $C \subseteq V$ με συνολικό κόστος

$$\sum_{v \in C} f(v) + \sum_{u \in V \setminus C} \min_{v \in C} \{\ell(u, v)\} \leq B \tag{1}$$

Σημείωση: Η αντιστοιχία με το πρόβλημα χωροθέτησης υπηρεσιών προκύπτει αν θεωρήσουμε ότι σε κάθε κορυφή $u \in V$ υπάρχει ένας πελάτης που πρέπει να συνδεθεί στην πλησιέστερη θέση όπου παρέχεται μια υπηρεσία. Οι αποστάσεις αντιστοιχούν στα κόστη σύνδεσης των πελατών. Το κόστος $f(v)$ αντιστοιχεί στο κόστος εγκατάστασης της υπηρεσίας στη θέση v. Επιλέγοντας να εγκαταστήσουμε την υπηρεσία στις κορυφές του C, έχουμε συνολικό κόστος εγκατάστασης $\sum_{v \in C} f(v)$, και κόστος σύνδεσης για κάθε κορυφή $u \in V \setminus C$ ίσο με $\min_{v \in C}\{\ell(u, v)\}$ (κάθε πελάτης συνδέεται στην πλησιέστερη θέση, το κόστος σύνδεσης για τους πελάτες στο C είναι μηδενικό). Το συνολικό κόστος του C δίνεται από την (1).

Υπόδειξη: Για τα τρία τελευταία προβλήματα, μπορείτε να δοκιμάσετε αναγωγές από το Κάλυμμα Κορυφών (Vertex Cover).

Οι επόμενες 4 ασκήσεις προτείνονται για εξάσκηση και βαθμολογούνται ως μπόνους!

Άσκηση 5*: Φωλιασμένα Διαστήματα

Δίνονται τα μη κενά κλειστά διαστήματα φυσικών αριθμών $[s_1, t_1], [s_2, t_2], \ldots, [s_n, t_n]$. Υποθέτουμε ότι $s_1 \leq s_2 \leq \cdots \leq s_n$, ενώ κάποια από τα s_i και t_i μπορεί να είναι ίδια. Δύο διαστήματα λέγονται φωλιασμένα όταν το ένα είναι *υποσύνολο* του άλλου. Το ζητούμενο είναι να υπολογίσουμε το *πλήθος* των ζευγών φωλιασμένων διαστημάτων στο $[s_1, t_1], [s_2, t_2], \dots, [s_n, t_n]$. Π.χ. στο $[1, 8], [2, 3], [3, 4], [5, 8], [6, 8], [7, 12]$ υπάρχουν 5 ζεύγη φωλιασμένων διαστημάτων, τα $[2, 3] \subseteq [1, 8]$, $[3, 4] \subseteq [1, 8]$, $[5, 8] \subseteq [1, 8]$, $[6, 8] \subseteq [1, 8]$, και $[6, 8] \subseteq [5, 8]$. Να διατυπώσετε ένα όσο το δυνατόν πιο αποδοτικό αλγόριθμο για αυτό το πρόβλημα. Να αιτιολογήσετε την ορθότητα και την χρονική πολυπλοκότητα του αλγορίθμου σας.

Άσκηση 6*: Το Πρόβλημα της Βιβλιοθήκης

Σε ένα φοιτητή δίνεται ένα υποχρεωτικό πρόγραμμα μελέτης για τις επόμενες T ημέρες. Με βάση το πρόγραμμα, ο φοιτητής διαβάζει ένα μόνο μάθημα κάθε ημέρα. Ο συνολικός αριθμός των μαθημάτων είναι n, $n < T$. Ο φοιτητής διαβάζει κάποια μαθήματα σε περισσότερες από μία και όχι κατ' ανάγκη συνεχόμενες ημέρες (π.χ. 1η μέρα Αλγόριθμοι, 2η Γλώσσες, 3η Βάσεις Δεδομένων, 4η Αλγόριθμοι, 5η Γλώσσες, 6η Αλγόριθμοι, 7η Αλγόριθμοι, 8η Βάσεις Δεδομένων, 9η Βάσεις Δεδομένων, κλπ.).

Η μελέτη κάθε μαθήματος απαιτεί το δανεισμό ενός συγκεκριμένου βιβλίου από τη βιβλιοθήκη. Ο κανονισμός δεν επιτρέπει στο φοιτητή να έχει "χρεωμένα" περισσότερα από k, $k < n$, βιβλία ταυτόχρονα. Επιπλέον, ο κανονισμός δεν επιτρέπει τον δανεισμό περισσότερων του ενός βιβλίου την ίδια ημέρα. Έτσι κάθε ημέρα που ο φοιτητής χρειάζεται ένα βιβλίο και δεν το έχει, πρέπει να πηγαίνει στη βιβλιοθήκη και να το δανείζεται (δεν μπορεί να δανειστεί και άλλα βιβλία την συγκεκριμένη ημέρα). Αν μάλιστα ο φοιτητής έχει ήδη "χρεωμένα" k βιβλία, πρέπει να διαλέξει ποιο βιβλίο θα επιστρέψει. Κάθε επίσκεψη στη βιβλιοθήκη αποσπά τον φοιτητή από τη μελέτη του. Το ζητούμενο λοιπόν είναι η πολιτική επιστροφής βιβλίων που ελαχιστοποιεί τον αριθμό των επισκέψεων στη βιβλιοθήκη.

Να διατυπώσετε έναν όσο το δυνατόν πιο αποδοτικό αλγόριθμο / πολιτική επιστροφής βιβλίων που εξασφαλίζει τον ελάχιστο αριθμό επισκέψεων στη βιβλιοθήκη, και να αποδείξετε ότι ο αλγόριθμός σας πράγματι υπολογίζει τη βέλτιστη λύση. Ποιός είναι ο χρόνος εκτέλεσης του αλγορίθμου σας; Μπορείτε να συνδέσετε το πρόβλημα της βιβλιοθήκης με κάποια (σημαντική) πρακτική εφαρμογή;

Άσκηση 7*: Κοπή Υφασμάτων

Δίνεται ένα ορθογώνιο κομμάτι υφάσματος με διαστάσεις $X \times Y$, όπου X και Y θετικοί ακέραιοι, και μια λίστα με n προϊόντα τα οποία μπορούν να κατασκευασθούν με την χρήση του υφάσματος. Για κάθε προϊόν i γνωρίζουμε ότι χρειάζεται ένα ορθογώνιο κομμάτι υφάσματος διαστάσεων $a_i \times b_i$, και ότι η τελική τιμή πώλησης είναι c_i (τα a_i, b_i, c_i είναι θετικοί ακέραιοι, αν σας διευκολύνει, μπορείτε να υποθέσετε ότι δεν υπάρχουν προϊόντα i, j με $a_i \geq a_j$, $b_i \geq b_j$, και $c_i \leq c_j$). Κάθε προϊόν μπορεί να δημιουργηθεί σε οσαδήποτε αντίγραφα (μπορεί και κανένα). Έχουμε στη διάθεσή μας μια μηχανή που μπορεί να κόψει οποιοδήποτε ορθογώνιο κομμάτι υφάσματος σε δύο κομμάτια, είτε οριζόντια είτε κατακόρυφα (κάθε κόψιμο ακολουθεί μια ευθεία κάθετη προς τις πλευρές του υφάσματος και δεν μπορεί να διακοπεί). Να διατυπώσετε έναν όσο το δυνατόν πιο αποδοτικό αλγόριθμο που προσδιορίζει την βέλτιστη εκμετάλλευση του υφάσματος, δηλ. υπολογίζει μια στρατηγική κοπής ώστε να μεγιστοποιηθεί το συνολικό κέρδος από την πώληση των προϊόντων που προκύπτουν από τα κομμάτια του υφάσματος. Να αιτιολογήσετε την ορθότητα και να προσδιορίσετε την χρονική πολυπλοκότητα του αλγορίθμου σας. *Σημείωση:* Πρόκειται για την Άσκηση **6.14** του DPV. Γενικά, το Κεφ. **6** (και όχι μόνο!) του DPV έχει πολύ ενδιαφέρουσες ασκήσεις.

Άσκηση 8*: Αντιπροσωπεία Φορητών Υπολογιστών

Επιθυμούμε να βελτιστοποιήσουμε την λειτουργία μιας νέας αντιπροσωπείας φορητών υπολογιστών για τις επόμενες n ημέρες. Για κάθε ημέρα i, $1 \leq i \leq n$, υπάρχει (ασφαλής) πρόβλεψη d_i του αριθμού των πωλήσεων. Όλες οι πωλήσεις λαμβάνουν χώρα το μεσημέρι, και όσοι φορητοί υπολογιστές δεν πωληθούν, αποθηκεύονται. Υπάρχει δυνατότητα αποθήκευσης μέχρι S υπολογιστών, και το κόστος είναι C για κάθε υπολογιστή που αποθηκεύεται και για κάθε ημέρα αποθήκευσης. Το κόστος μεταφοράς για την προμήθεια νέων υπολογιστών είναι K ευρώ, ανεξάρτητα από το πλήθος των υπολογιστών που προμηθευόμαστε, και οι νέοι υπολογιστές φθάνουν λίγο πριν το μεσημέρι (άρα αν πωληθούν αυθημερόν, δεν χρειάζονται αποθήκευση). Αρχικά δεν υπάρχουν καθόλου υπολογιστές στην αντιπροσωπεία.

Το ζητούμενο είναι να προσδιορισθούν οι παραγγελίες (δηλ. πόσους υπολογιστές θα παραγγείλουμε και πότε) ώστε να ικανοποιηθούν οι προβλεπόμενες πωλήσεις με το ελάχιστο δυνατό συνολικό κόστος (αποθήκευσης και μεταφοράς). Να διατυπώσετε αλγόριθμο με χρόνο εκτέλεσης πολυωνυμικό στο nS για αυτό το πρόβλημα. Να αιτιολογήσετε την ορθότητα και την πολυπλοκότητα του αλγορίθμου σας.