3η Σειρά Ασκήσεων

- Να επιλύσετε τρία από τα πέντε (ισοδύναμα) προβλήματα. Παράδοση: 13 Μαρτίου 2009 (στην έναρξη της διάλεξης).

1. **[Διασκευή του Προβλήματος 1.6 από το βιβλίο Algorithm Design των Kleinberg και Tardos.]** Μία εταιρεία θαλασσίων μεταφορών κατέχει n πλοία και παρέχει υπηρεσίες σε n λιμάνια. Καθένα από τα πλοία της έχει ένα χρονοδιάγραμμα το οποίο προσδιορίζει για καθεμιά από τις επόμενες m μέρες, όπου $m > n$, αν το πλοίο θα βρίσκεται σε ταξίδι και ποιο λιμάνι θα επισκεφτεί (αν βρίσκεται σε ταξίδι). Κάθε πλοίο επισκέπτεται ένα λιμάνι για μία μόνο μέρα και ικανοποιεί την εξής συνθήκη ασφαλείας: Δεν μπορούν δύο πλοία να βρίσκονται στο ίδιο λιμάνι την ίδια μέρα.

   Η εταιρεία προγραμματίζει μία συντήρηση των πλοίων της στις επόμενες m μέρες κατά τον ακόλουθο τρόπο. Το χρονοδιάγραμμα κάθε πλοίου θα περικοπεί. Συγκεκριμένα, για κάθε πλοίο, θα υπάρχει κάποια μέρα κατά την οποία το πλοίο θα φτάσει στο συγκεκριμένο λιμάνι και θα παραμείνει εκεί (για συντήρηση) κατά τις υπόλοιπες μέρες. Αυτό σημαίνει ότι το πλοίο δεν θα επισκεφτεί τα υπόλοιπα λιμάνια του χρονοδιαγράμματός του (αν τέτοια υπάρχουν). Έτσι, η περικοπή του χρονοδιαγράμματος του πλοίου αποτελείται απλά από το αρχικό του χρονοδιάγραμμα μέχρι κάποια καθορισμένη μέρα, όπου το πλοίο φτάνει σε κάποιο λιμάνι στο οποίο θα παραμείνει για τις υπόλοιπες μέρες.

   Η εταιρεία θέλει να βρει μία περικοπή κάθε χρονοδιαγράμματος έτσι ώστε η συνθήκη ασφαλείας να εξακολουθήσει να ισχύει.

   (α) Αποδείξτε ότι υπάρχει πάντοτε ένα τέτοιο σύνολο περικοπών.

   (β) Παρουσιάστε και αναλύστε ένα αποδοτικό αλγόριθμο για την εύρεση ενός τέτοιου συνόλου περικοπών.

2. **[Διασκευή του Προβλήματος 1.7 από το βιβλίο Algorithm Design των Kleinberg και Tardos.]** Ως εργαζόμενοι σε μία κατασκευαστική εταιρεία μεγάλων δικτύων επικοινωνίας, αναζητούμε αλγορίθμους *μεταγωγής εισόδου/εξόδου* σε ένα *ετεροσυζεύκτη* (crossbar). Συγκεκριμένα, υπάρχουν n *αγωγοί εισόδου* και n *αγωγοί εξόδου,* και ο κάθε αγωγός κατευθύνεται από μία *αφετηρία* σε ένα *προορισμό.* Ο αγωγός εισόδου i αντιστοιχεί στη *είσοδο* i, ενώ ο αγωγός εξόδου j αντιστοιχεί στην *έξοδο* j, όπου $1 \leq i, j \leq n$. Κάθε αγωγός εισόδου συναντά κάθε αγωγό εξόδου σε μόνο ένα διακριτό σημείο, σε μία ειδική συσκευή του υλικού που ονομάζεται *κιβώτιο ένωσης.* Τα σημεία στο ίδιο καλώδιο είναι φυσικά διατεταγμένα από την πηγή προς τον προορισμό. Για δύο διαφορετικά σημεία x και y πάνω στο ίδιο καλώδιο, λέμε ότι το x *προηγείται* του y (και ότι το y έπεται του x) αν το x βρίσκεται πλησιέστερα προς την πηγή από το y. Η σειρά με την οποία ένα συγκεκριμένο καλώδιο εισόδου συναντά τα καλώδια εξόδου δυνατόν να διαφέρει από τη σειρά με την οποία ένα άλλο καλώδιο εισόδου συναντά τα καλώδια εξόδου, και το

αντίστοιχο ισχύει για τη σειρά με την οποία ένα συγκεκριμένο καλώδιο εξόδου συναντά τα καλώδια εισόδου.

Κάθε καλώδιο εισόδου φέρει κάποιο *ρεύμα εισόδου* το οποίο χρειάζεται να μεταγωγηθεί σε κάποιο από τα καλώδια εξόδου. Αν το ρεύμα της εισόδου i μεταγωγηθεί στοην έξοδο j, στο κιβώτιο ένωσης B, τότε το ρεύμα διασχίζει το καλώδιο εισόδου i από την είσοδο i μέχρι το κιβώτιο ένωσης B, όπου μεταγωγείται στο καλώδιο εξόδου j. Από εκεί, διασχίζει το καλώδιο εξόδου j μέχρι την έξοδο j. Δεν είναι σημαντικό ποιο ρεύμα εισόδου μεταγωγείται σε ποια έξοδο, αλλά είναι απαραίτητο κάθε συγκεκριμένο ρεύμα εισόδου να μεταγωγηθεί σε μία διαφορετική έξοδο από οποιοδήποτε άλλο ρεύμα εισόδου. Η δυσκολότερη απαίτηση, ωστόσο, είναι ότι δεν επιτρέπεται να περάσουν από το ίδιο κιβώτιο ένωσης δύο διαφορετικά ρεύματα εισόδου μετά τη μεταγωγή. 'Ετσι, μία μεταγωγή είναι *έγκυρη* αν κάθε ρεύμα εισόδου μεταγωγείται σε διαφορετική έξοδο, και ανά δύο τα ρεύματα που προκύπτουν από τη μεταγωγή δεν περνούν από το ίδιο κιβώτιο ένωσης.

(α) Αποδείξτε ότι υπάρχει πάντοτε μία έγκυρη μεταγωγή.

(β) Παρουσιάστε και αναλύστε ένα αποδοτικό αλγόριθμο για την εύρεση μίας έγκυρης μεταγωγής.

3. **[Διασκευή του Προβλήματος 1.8 από το βιβλιο Algorithm Design των Kleinberg και Tardos.]** Επανεξετάζουμε τον αλγόριθμο Gale-Shapley αναθεωρώντας την υπόθεση ότι οι άνδρες και οι γυναίκες είναι ειλικρινείς ως προς τη σειρά των προτιμήσεών τους. Το βασικό ερώτημα είναι κατά πόσο ένας άνδρας ή μία γυναίκα μπορεί να πετύχει καλύτερο αποτέλεσμα αν πει ψέματα για τη σειρά των προτιμήσεων που έχει.

   Συγκεκριμένα, υποθέτουμε ότι κάθε άνδρας και κάθε γυναίκα έχει μία πραγματική σειρά προτιμήσεων. Θεωρούμε τώρα μία γυναίκα w. Υποθέτουμε ότι η w προτιμά τον άνδρα m από τον άνδρα m', αλλά και οι δύο βρίσκονται σχετικά χαμηλά στη λίστα των προτιμήσεών της. Μπορεί άραγε με εναλλαγή της σειράς των m και m' στη λίστα των προτιμήσεών της και εκτέλεση του αλγορίθμου Gale-Shapley με αυτή τη ψευδή λίστα προτιμήσεων (για τη γυναίκα w), η w να πετύχει να καταλήξει με έναν άνδρα m'' τον οποίο προτιμά έναντι και του m και του m' στην πραγματική λίστα των προτιμήσεών της; Απαντείστε αυτή την ερώτηση με έναν από τους ακόλουθους δύο τρόπους:

   - Αποδείξτε ότι για κάθε σύνολο από λίστες προτιμήσεων, η εναλλαγή της σειράς δύο ανδρών στη λίστα προτιμήσεων μίας γυναίκας δεν μπορεί να βελτιώσει το σύντροφο της γυναίκας που θα αποδώσει ο αλγόριθμος Gale-Shapley.
   - Δώστε ένα αντιπαράδειγμα συνόλου προτιμήσεων για το οποίο υπάρχει εναλλαγή της σειράς δύο ανδρών στη λίστα προτιμήσεων μίας γυναίκας που θα μπορούσε να βελτιώσει το σύντροφό της (που θα αποδώσει ο αλγόριθμος Gale-Shapley).

4. **[Διασκευή του Προβλήματος 4.7 από το βιβλιο Algorithm Design των Kleinberg και Tardos.]** Μία μηχανή αναζήτησης έχει να εκτελέσει σημαντικούς υπολογισμούς κάθε

φορά που ανασυγκροτεί το ευρετήριό της. Ευτυχώς, η μηχανή διαθέτει ένα μεγάλο υπερϋπολογιστή, μαζί με ένα ουσιαστικά απεριόριστο αριθμό από προσωπικούς υπολογιστές υψηλής τεχνολογίας. Το σύνολο των υπολογισμών έχει χωριστεί σε n διακριτές και ανεξάρτητες εργασίες $J_1, \ldots, J_n$. Κάθε εργασία J_i, όπου $1 \leq i \leq n$, εκτελείται σε δύο διαδοχικές φάσεις. Η *προεπεξεργασία* χρειάζεται χρόνο p_i και εκτελείται στον υπερϋπολογιστή, ενώ η *ολοκλήρωση* χρειάζεται χρόνο f_i και λάμβάνει χώρα σε κάποιο προσωπικό υπολογιστή.

Επειδή υπάρχουν διαθέσιμοι τουλάχιστον n προσωπικοί υπολογιστές, όλες οι ολοκληρώσεις μπορούν να γίνουν ταυτόχρονα. Ωστόσο, ο υπερϋπολογιστής μπορεί να εκτελεί μόνο μία εργασία κάθε φορά. Έτσι, χρειάζεται να βρούμε τη σειρά εκτέλεσης της προεπεξεργασίας των εργασιών στον υπερϋπολογιστή. Κάθε φορά που τελειώνει η προεπεξεργασία μίας εργασίας στον υπερϋπολογιστή, αυτή μπορεί να περάσει σε κάποιο προσωπικό υπολογιστή, ενώ μία άλλη εργασία μπορεί να ανατεθεί στον υπερϋπολογιστή.

Το *χρονοδιάγραμμα* είναι μία διάταξη των εργασιών στον υπερϋπολογιστή, και ο *χρόνος ολοκλήρωσης* του χρονοδιαγράμματος είναι ο συντομότερος χρόνος στον οποίο θα έχει ολοκληρωθεί η επεξεργασία όλων των εργασιών στους προσωπικούς υπολογιστές. Παρουσιάστε και αναλύστε ένα αποδοτικό αλγόριθμο ο οποίος θα βρίσκει ένα χρονοδιάγραμμα με τον ελάχιστο δυνατό χρόνο ολοκλήρωσης.

5. [Διασκευή του Προβλήματος 4.13 από το βιβλίο Algorithm Design των Kleinberg και Tardos.] Μία μικρή επιχείρηση δέχεται n εργασίες τις οποίες πρέπει να εκτελέσει στο μοναδικό μηχάνημά της. Η εργασία i, όπου $1 \leq i \leq n$, έχει διάρκεια t_i και βάρος w_i. Ένα *χρονοδιάγραμμα* προσδιορίζει τη σειρά εκτέλεσης των εργασιών στο μηχάνημα. Επομένως, ένα χρονοδιάγραμμα προσδιορίζει το *χρόνο ολοκλήρωσης* C_i κάθε εργασίας i. Αν η εργασία i εκτελεστεί αμέσως μετά από την εργασία j, τότε $C_i = C_j + t_i$. (Αν η εργασία i εκτελεστεί πρώτη, τότε $C_i = t_i$.)

   Παρουσιάστε και αναλύστε ένα αποδοτικό αλγόριθμο για την εύρεση ενός χρονοδιαγράμματος το οποίο ελαχιστοποιεί το σταθμισμένο άθροισμα $\sum_{i=1}^{n} w_i C_i$.