Λύσεις 3ης Σειράς Ασκήσεων

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.

Λύση: Με δεδομένο ένα χρονοδιάγραμμα, πρέπει να επιλέξουμε για κάθε πλοίο ένα σταθμό — το λιμάνι στο οποίο το πλοίο θα αγκυροβολήσει (όταν φτάσει) για το υπόλοιπο των n ημερών. Η συνθήκη ασφαλείας συνεπάγεται ότι όλοι οι σταθμοί που θα επιλέξουμε πρέπει να είναι μεταξύ τους διαφορετικοί. Κατσκευάζουμε από το χρονοδιάγραμμα ένα στιγμιότυπο του προβλήματος του ευσταθούς ταιριάσματος ως ακολούθως:

- Κάθε πλοίο κατατάσσει τα λιμάνια κατά τη χρονολογική σειρά των επίσκέψεών του (στα λιμάνια).
- Κάθε λιμάνι κατατάσσει τα πλοία κατά την αντίστροφη χρονολογική σειρά των επισκέψεών τους (στο λιμάνι).

Προφανώς, κάθε ταίριασμα μεταξύ πλοίων και λιμανιών (για το στιγμιότυπο που κατασκευάσαμε) ορίζει ένα σύνολο περικοπών: κάθε πλοίο αγκυροβολεί στο λιμάνι με το οποίο είναι ταιριασμένο. Θα αποδείξουμε ότι κάθε ευσταθές ταίριασμα ορίζει ένα χρονοδιάγραμμα που ικανοποιεί τη συνθήκη ασφαλείας:

> Υποθέτουμε, για να καταλήξουμε σε αντίφαση, ότι το χρονοδιάγραμμα που ορίζεται από ένα ευσταθές ταίριασμα δεν ικανοποιεί τη συνθήκη ασφαλείας. Δηλαδή, υπάρχει κάποιο πλοίο S_i το οποίο περνά από κάποιο λιμάνι P_k (κατά ή) μετά την αγκυροβόληση κάποιου άλλου πλοίου S_j στο ίδιο λιμάνι.
>
> - Αφού δεν υπάρχουν δύο πλοία που φτάνουν στο ίδιο λιμάνι την ίδια μέρα, έπεται ότι το πλοίο S_i φτάνει στο λιμάνι P_k αυστηρά μετά την άφιξη του πλοίου

S_j στο λιμάνι P_k. Από την κατασκευή μας, έπεται ότι το λιμάνι P_k προτιμάει το πλοίο S_i έναντι του πλοίου S_j με το οποίο είναι ταιριασμένο.

- Από τον ορισμό του ταιριάσματος, το λιμάνι στο οποίο αγκυροβολεί το πλοίο S_i είναι διαφορετικό από το λιμάνι P_k (αφού αυτό είναι το λιμάνι με το οποίο το πλοίο S_i ταιριάζεται). Έστω ότι αυτό είναι το λιμάνι P_l (με $l \neq k$). Αφού στο χρονοδιάγραμμα που κατασκευάζουμε κανένα πλοίο δεν επισκέπτεται άλλα λιμάνια μετά την αγκυροβόλησή του, έπεται ότι το πλοίο S_i φτάνει στο λιμάνι P_l μετά το πέρασμά του από το λιμάνι P_k. Από την κατασκευή μας, έπεται ότι το πλοίο S_i προτιμάει το λιμάνι P_k έναντι του λιμανιού P_l με το οποίο είναι ταιριασμένο.

Άρα, το πλοίο S_i και το λιμάνι P_k αποτελούν επικίνδυνο ζευγάρι. Αντίφαση, αφού ξεκινήσαμε με ένα ευσταθές ταίριασμα.

Ο αλγόριθμος των Gale και Shapley συνεπάγεται ότι υπάρχει ένα ευσταθές ταίριασμα. Έπεται ότι υπάρχει και ένα χρονοδιάγραμμα που ικανοποιεί τη συνθήκη ασφαλείας.

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

Λύση:

Ο αλγόριθμος: Από το στιγμιότυπο του προβλήματός μας, κατασκευάζουμε ένα στιγμιότυπο του προβλήματος του ευσταθούς ταιριάσματος (όπως παραπάνω). Χρησιμοποιούμε τον αλγόριθμο των Gale και Shapley για να υπολογίσουμε ένα ευσταθές ταίριασμα. Ο σταθμός κάθε πλοίου θα είναι το λιμάνι με το οποίο ταιριάζεται.

Πολυπλοκότητα: Η κατασκευή του στιγμιοτύπου του προβλήματος του ευσταθούς ταιριάσματος χρειάζεται $O(n^2)$ βήματα αφού έχουμε να κατασκευάσουμε n λίστες προτίμησης με n στοιχεία η καθεμιά. Αφού η πολυπλοκότητα του αλγορίθμου των Gale και Shapley είναι $O(n^2)$, έπεται ότι η πολυπλοκότητα του αλγορίθμου μας είναι επίσης $O(n^2)$.

2. [Διασκευή του Προβλήματος 1.7 από το βιβλιο 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 στον ίδιο αγωγό είναι φυσικά διατεταγμένα από την αφετηρία προς τον προορισμό. Για δύο διαφορετικά σημεία x και y πάνω στον ίδιο αγωγό, λέμε ότι το x *προηγείται* του y (και ότι το y έπεται του x) αν το x βρίσκεται πλησιέστερα προς την αφετηρία από το y. Η σειρά με την οποία ένας συγκεκριμένος αγωγός εισόδου συναντά τους αγωγούς εξόδου δυνατόν να διαφέρει από τη σειρά με την οποία ένας άλλος αγωγός εισόδου συναντά τους αγωγούς εξόδου, και το αντίστοιχο ισχύει για τη σειρά με την οποία ένας συγκεκριμένος αγωγός εξόδου συναντά τους αγωγούς εισόδου.

Κάθε αγωγός εισόδου φέρει κάποιο *ρεύμα εισόδου* το οποίο χρειάζεται να μεταγωγηθεί σε κάποιο από τους αγωγούς εξόδου. Αν το ρεύμα της εισόδου i μεταγωγηθεί στοην έξοδο j, στο κιβώτιο ένωσης B, τότε το ρεύμα διασχίζει τον αγωγό εισόδου i από την είσοδο i μέχρι το κιβώτιο ένωσης B, όπου μεταγωγείται στον αγωγό εξόδου j. Από εκεί, διασχίζει τον αγωγό 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. Έτσι, μία μεταγωγή είναι *<u>έγκυρη</u>* 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.

<u>Λύση:</u> Με δεδομένο ένα ετεροσυζεύκτη, πρέπει να επιλέξουμε για κάθε αγωγό εισόδου έναν αγωγό εξόδου – εκείνον στον οποίο θα μεταγωγηθεί. Οι αγωγοί εξόδου που θα επιλέξουμε πρέπει να είναι όλοι μεταξύ τους διαφορετικοί. Κατασκευάζουμε από τον ετεροσυζεύκτη ένα στιγμιότυπο του προβλήματος του ευσταθούς ταιριάσματος ως ακολούθως:

- Κάθε αγωγός εισόδου κατατάσσει τους αγωγούς εξόδου κατά τη σειρά με την οποία τους συναντά (από την πηγή προς τον προορισμό).
- Κάθε αγωγός εξόδου κατατάσσει τους αγωγούς εισόδου κατά την αντίστροφη σειρά προς εκείνη με την οποία τους συναντά (από την πηγή προς τον προορισμό).

Προφανώς, κάθε ταίριασμα μεταξύ αγωγών εισόδου και αγωγών εξόδου (για το στιγμιότυπο που κατασκευάσαμε) ορίζει μία μεταγωγή: κάθε αγωγός εισόδου μεταγωγείται στον αγωγό εξόδου με το οποίο είναι ταιριασμένος. Θα αποδείξουμε ότι κάθε ευσταθές ταίριασμα ορίζει μία έγκυρη μεταγωγή:

> Υποθέτουμε, για να καταλήξουμε σε αντίφαση, ότι η μεταγωγή που ορίζεται από ένα ευσταθές ταίριασμα δεν είναι έγκυρη. Τότε, υπάρχει κάποιος αγωγός

εξόδου i το οποίο συναντά κάποιο αγωγό εξόδου k (κατά ή) μετά τη μεταγώγηση κάποιου άλλου αγωγού εισόδου j στον ίδιο αγωγό εξόδου.

- Αφού δεν υπάρχουν δύο αγωγοί εισόδου που συναντούν τον ίδιο αγωγό εξόδου στο ίδιο σημείο, έπεται ότι ο αγωγός εισόδου i συναντά τον αγωγό εξόδου k μετά το σημείο στο οποίο ο αγωγός εισόδου j συναντά τον αγωγό εξόδου k. Από την κατασκευή μας, έπεται ότι ο αγωγός εξόδου k προτιμάει τον αγωγό εισόδου i έναντι του αγωγού εισόδου j με τον οποίο ταιριάστηκε.
- Από τον ορισμό του ταιριάσματος, ο αγωγός εξόδου στον οποίο μεταγωγείται ο αγωγός εισόδου i είναι διαφορετικός από τον αγωγό εξόδου k (αφού αυτός είναι ο αγωγός εξόδου με τον οποίο ο αγωγός εισόδου i ταιριάζεται). Έστω ότι αυτός είναι ο αγωγός εξόδου l (με $l \neq k$). Αφού στη μεταγωγή που κατασκευάζουμε κανένας αγωγός εισόδου δεν συναντά άλλους αγωγούς εξόδου μετά την μεταγωγή του, έπεται ότι ο αγωγός εισόδου i συναντά τον αγωγό εξόδου l μετά τη συνάντησή του με τον αγωγό εξόδου k. Από την κατασκευή μας, έπεται ότι ο αγωγός εισόδου i προτιμάει τον αγωγό εξόδου k έναντι του αγωγού εξόδου l με τον οποίο ταιριάστηκε.

Άρα, ο αγωγός εισόδου i και ο αγωγός εξόδου k αποτελούν επικίνδυνο ζευγάρι. Αντίφαση, αφού ξεκινήσαμε με ένα ευσταθές ταίριασμα.

Ο αλγόριθμος των Gale και Shapley συνεπάγεται ότι υπάρχει ένα ευσταθές ταίριασμα. Έπεται ότι υπάρχει και μία έγκυρη μεταγωγή.

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

Λύση:

Ο αλγόριθμος: Από το στιγμιότυπο του προβλήματός μας, κατασκευάζουμε ένα στιγμιότυπο του προβλήματος του ευσταθούς ταιριάσματος (όπως παραπάνω). Χρησιμοποιούμε τον αλγόριθμο των Gale και Shapley για να υπολογίσουμε ένα ευσταθές ταίριασμα. Ο αγωγός εξόδου στον οποίο μεταγωγείται κάθε αγωγός εισόδου θα είναι ο αγωγός εξόδου με τον οποίο ταιριάζεται.

Πολυπλοκότητα: Η κατασκευή του στιγμιοτύπου του προβλήματος του ευσταθούς ταιριάσματος χρειάζεται $O(n^2)$ βήματα αφού έχουμε να κατασκευάσουμε n λίστες προτίμησης με n στοιχεία η καθεμιά. Αφού η πολυπλοκότητα του αλγορίθμου των Gale και Shapley είναι $O(n^2)$, έπεται ότι η πολυπλοκότητα του αλγορίθμου μας είναι επίσης $O(n^2)$.

3. [Διασκευή του Προβλήματος 1.8 από το βιβλιο Algorithm Design των Kleinberg και Tardos.] Επανεξετάζουμε τον αλγόριθμο των Gale και 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 των Gale και 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 των Gale και Shapley).

Λύση: Θα δώσουμε ένα αντιπαράδειγμα. Έστω ένα στιγμιότυπο του προβλήματος του ευσταθούς ταιριάσματος με τρεις άνδρες m_1, m_2 και m_3 και τρεις γυναίκες w_1, w_2 και w_3. Οι σειρές προτιμήσεων φαίνονται στον ακόλουθο πίνακα:

m_1	m_2	m_3	w_1	w_2	w_3	w_3'
w_3	w_1	w_3	m_1	m_1	m_2	m_2
w_1	w_3	w_1	m_2	m_2	m_1	m_3
w_2	w_2	w_2	m_3	m_3	m_3	m_1

Η τελευταία στήλη δείχνει τις ψευδείς προτιμήσεις της γυναίκας w_3. (Ένας ισοδύναμος τρόπος να το δούμε είναι να θεωρήσουμε ότι αυτές είναι οι πραγματικές προτιμήσεις μίας άλλης γυναίκας w_3' με την οποία αντικαθιστούμε τη γυναίκα w_3.)

Ας δούμε πρώτα μία εκτέλεση του αλγορίθμου των Gale και Shapley με τις αληθείς προτιμήσεις της γυναίκας w_3. Οι τρεις άνδρες προτείνουν με τη σειρά m_1, m_2, m_3. Η εκτέλεση αυτή θα δημιουργήσει το ζευγάρωμα $\{(m_1, w_3), (m_2, w_1), (m_3, w_2)\}$.

Ας δούμε τώρα μία εκτέλεση του αλγορίθμου των Gale και Shapley με τις ψευδείς προτιμήσεις της γυναίκας w_3 (ή με τις αληθείς προτιμήσεις της γυναίκας w_3'). Οι τρεις άνδρες αρχίζουν να προτείνουν με τη σειρά m_1, m_2, m_3. Αρχικά, δημιουργούνται τα ζεύγη (m_1, w_3') και (m_2, w_1). Όταν ο άνδρας m_3 προτείνει στη γυναίκα w_3', και ο άνδρας m_1 θα απελευθερωθεί, οπότε θα δημιουργηθεί το ζευγάρι (m_3, w_3') και ο άνδρας m_1 θα είναι ο μοναδικός ελεύθερος άνδρας. Τότε, ο άνδρας m_1 θα προτείνει στη γυναίκα w_1, οπότε θα δημιουργηθεί το ζευγάρι (m_1, w_1)

και θα απελευθερωθεί ο άνδρας m_2. Στη συνέχεια, ο άνδρας m_2 θα προτείνει στη γυναίκα w'_3, η οποία θα δεχτεί, οπότε θα δημιουργηθεί το ζευγάρι (m_2, w'_3) και θα απελευθερωθεί ο άνδρας m_3. Στη συνέχεια, ο άνδρας m_3 προτείνει στην (ακόμη ελεύθερη) γυναίκα w_2, οπότε δημιουργείται το ζευγάρι (m_3, w_2). 'Ετσι, η εκτέλεση του αλγορίθμου των Gale και Shapley με τις ψευδείς προτιμήσεις της γυναίκας w'_3 θα δημιουργήσει το ζευγάρωμα $\{(m_1, w_1), (m_2, w'_3), (m_3, w_2)\} = \{(m_1, w_1), (m_2, w_3), (m_3, w_2)\}$.

Παρατηρούμε ότι η γυναίκα w_3 είναι περισσότερο ικανοποιημένη στο ζευγάρωμα που παράγει η εκτέλεση του αλγορίθμου των Gale και Shapley με τις ψευδείς προτιμήσεις της παρά στο ζευγάρωμα που παράγει η εκτέλεση του αλγορίθμου των Gale και Shapley με τις αληθείς προτιμήσεις της

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. 'Ετσι, 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.

Λύση: Παρατηρούμε ότι η προεπεξεργασία όλων των εργασιών στον υπερϋπολογιστή τελειώνει σε χρόνο που είναι ανεξάρτητος του χρονοδιαγράμματος. (Αυτός ο χρόνος δεν είναι παρά το άθροισμα $\sum_{i=1}^{n} p_i$ των χρόνων προεπεξεργασίας των εργασιών.) Αφού θέλουμε να ελαχιστοποιήσουμε τον χρόνο ολοκλήρωσης, είναι φυσιολογικό να θέλουμε η τελευταία εργασία που θα στείλουμε σε κάποιο προσωπικό υπολογιστή να έχει τον ελάχιστο δυνατό χρονο ολοκλήρωσης f_i, όπου $1 \leq i \leq n$. Η αρχική αυτή παρατήρηση θα είναι η βάση για τον άπληστο αλγόριθμο που θα σχεδιάσουμε.

Ο άπληστος αλγόριθμος: Παρουσιάζουμε τον εξής άπληστο αλγόριθμο:

Χρονοδρομολογούμε τις εργασίες κατά μη αύξοντα χρόνο ολοκλήρωσης f_i, όπου $1 \leq i \leq n$.

Προκαταρκτικά και τεχνικές: Θα χρησιμοποιήσουμε την τεχνική επιχειρήματος ανταλλαγής για να αποδείξουμε τη βελτιστότητα του απλήστου αλγορίθμου (ότι, δηλαδή, ο άπληστος αλγόριθμος κατασκευάζει ένα χρονοδιάγραμμα το οποίο ελαχιστοποιεί τον χρόνο ολοκλήρωσης). Έστω O το χρονοδιάγραμμα που κατασκευάζει ο άπληστος αλγόριθμος, και B το βέλτιστο χρονοδιάγραμμα (το οποίο κατασκευάζει ο βέλτιστος αλγόριθμος). Υποθέτουμε ότι τα χρονοδιαγράμματα O και B είναι διαφορετικά (αλλοιώς, το O είναι βέλτιστο και έχουμε τελειώσει). Ο σκοπός μας είναι να μετασχηματίσουμε διαδοχικά το βέλτιστο χρονοδιάγραμμα B στο άπληστο χρονοδιάγραμμα O. Θα χρησιμοποιήσουμε για τούτο μία ακολουθία ανταλλαγών καθεμιά από τις οποίες ανταλλάσσει δύο εργασίες στο χρονοδιάγραμμα B οι οποίες εκτελούνται με διαφορετική σειρά στο χρονοδιάγραμμα O. Το σημαντικό είναι να δείξουμε ότι κάθε ανταλλαγή έχει ως αποτέλεσμα την μη αύξηση του χρόνου ολοκλήρωσης. Αυτό συνεπάγεται ότι το τελικό χρονοδιάγραμμα που θα προκύψει από τους μετασχηματισμούς μας δεν έχει μεγαλύτερο χρόνο ολοκλήρωσης από το αρχικό χρονοδιάγραμμα B. Αφού το αρχικό χρονοδιάγραμμα B ήταν βέλτιστο, έπεται ότι και το τελικό χρονοδιάγραμμα θα είναι βέλτιστο. Αφού το τελικό αυτό χρονοδιάγραμμα ταυτίζεται με το άπληστο χρονοδιάγραμμα O, έπεται ότι το άπληστο χρονοδιάγραμμα O είναι επίσης βέλτιστο. Προχωράμε με τις τυπικές λεπτομέρειες της απόδειξης.

Η απόδειξη βελτιστότητας με επιχείρημα ανταλλαγής: Αφού τα χρονοδιαγράμματα O και B είναι διαφορετικά, υπάρχει ζεύγος εργασιών i και j τέτοιες ώστε η j προηγείται της i στο χρονοδιάγραμμα O, ενώ η i προηγείται της j στο χρονοδιάγραμμα B. Χωρίς βλάβη της γενικότητας, μπορούμε να υποθέσουμε ότι οι εργασίες i και j είναι διαδοχικές στο χρονοδιάγραμμα B. (Αν όλα τα ζεύγη διαδοχικών εργασιών στο χρονοδιάγραμμα O είχαν την ίδια σειρά και στο χρονοδιάγραμμα B, τότε τα χρονοδιαγράμματα O και B θα ήταν τα ίδια και το χρονοδιάγραμμα O θα ήταν βέλτιστο.) Από τον ορισμό του απλήστου αλγορίθμου, έχουμε ότι $f_j \geq f_i$. Έστω S_i και S_j οι χρόνοι περάτωσης της προεπεξεργασίας των εργασιών i και j, αντίστοιχα, στον υπερϋπολογιστή για το χρονοδιάγραμμα B. Έστω P_i και P_j οι χρόνοι περάτωσης της ολοκλήρωσης των εργασιών i και j, αντίστοιχα, στους προσωπικούς υπολογιστές για το χρονοδιάγραμμα B. Προφανώς, $P_i = S_i + f_i$ και $P_j = S_j + f_j$.

Ανταλλάσσουμε τις εργασίες i και j στο χρονοδιάγραμμα B. Έστω $\overline{B}$ το χρονοδιάγραμμα που προκύπτει. Παρατηρούμε ότι οι χρόνοι περάτωσης της ολοκλήρωσης όλων των εργασιών πλην των i και j παραμένουν οι ίδιοι στο χρονοδιάγραμμα $\overline{B}$ όπως ήταν στο χρονοδιάγραμμα B. Επομένως, μας ενδιαφέρουν μόνο οι νέοι χρόνοι (περάτωσης της) ολοκλήρωσης των εργασιών i και j. Έστω $\overline{P}_i$ και $\overline{P}_j$ οι χρόνοι περάτωσης της ολοκλήρωσης των εργασιών i και j, αντίστοιχα, στους προσωπικούς υπολογιστές για το χρονοδιάγραμμα B. Από την κατασκευή μας, $\overline{P_i} = S_j + f_i$ και $\overline{P_j} = S_i + f_j$.

Προφανώς, ο χρόνος (περάτωσης της) ολοκλήρωσης της εργασίας j μειώνεται στο χρονοδιάγραμμα $\overline{B}$ ενώ ο χρόνος (περάτωσης της) ολοκλήρωσης της εργασίας i αυξάνεται. Έτσι, χρειάζεται μόνο να εξετάσουμε την επίδραση του νέου χρόνου (περάτωσης της) ολοκλήρωσης της εργασίας i στον χρόνο ολοκλήρωσης του χρονοδιαγράμματος $\overline{B}$. Προφανώς, $\overline{P}_i = S_j + f_i \leq S_j + f_j = P_j$. Έπεται ότι ο χρόνος ολοκλήρωσης του χρονοδιαγράμματος $\overline{B}$ δεν είναι μεγαλύτερος από τον χρόνο ολοκλήρωσης του χρονοδιαγράμματος B. Αφού το χρονοδιάγραμμα B ήταν βέλτιστο, έπεται ότι και το χρονοδιάγραμμα $\overline{B}$ είναι βέλτιστο. Αφού το τελικό χρονοδιάγραμμα $\overline{B}$ ταυτίζεται με το άπληστο χρονοδιάγραμμα O, έπεται ότι το άπληστο χρονοδιάγραμμα O είναι βέλτιστο.

Πολυπλοκότητα: Η πολυπλοκότητα του αλγόριθμου προέρχεται από την ταξινόμηση των χρόνων ολοκλήρωσης f_i των εργασιών, όπου $1 \leq i \leq n$, και είναι $\Theta(n \lg n)$.

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 $\sum_{i=1}^{n} w_i C_i$.

Λύση: Διαισθητικά, εργασίες με μεγάλα βάρη και μικρή διάρκεια πρέπει να εκτελούνται νωρίτερα, αφού αυτές μπορούν να συνεισφέρουν περισσότερο στο σταθμισμένο άθροισμα $\sum_{i=1}^{n} w_i C_i$ (λόγω του μεγάλου βάρους που έχουν), και επομένως θα ήταν καλό να πολλαπλασιάσουν ένα όσο γίνεται πιο μικρό χρόνο ολοκλήρωσης. Η αρχική αυτή παρατήρηση θα είναι η βάση για τον άπληστο αλγόριθμο που θα σχεδιάσουμε.

Ο άπληστος αλγόριθμος: Παρουσιάζουμε τον εξής άπληστο αλγόριθμο:

Χρονοδρομολογούμε τις εργασίες κατά μη αύξοντα λόγο $\frac{w_i}{t_i}$, όπου $1 \leq i \leq n$.

Προκαταρκτικά και τεχνικές: Θα χρησιμοποιήσουμε την τεχνική επιχειρήματος ανταλλαγής για να αποδείξουμε τη βελτιστότητα του απλήστου αλγορίθμου (ότι, δηλαδή, ο άπληστος αλγόριθμος κατασκευάζει ένα χρονοδιάγραμμα το οποίο ελαχιστοποιεί το σταθμισμένο άθροισμα $\sum_{i=1}^{n} w_i C_i$).

Έστω O το χρονοδιάγραμμα που κατασκευάζει ο άπληστος αλγόριθμος, και B το βέλτιστο χρονοδιάγραμμα (το οποίο κατασκευάζει ο βέλτιστος αλγόριθμος). Υποθέτουμε ότι τα χρονοδιαγράμματα O και B είναι διαφορετικά (αλλοιώς, το O είναι βέλτιστο και έχουμε τελειώσει). Ο σκοπός μας είναι να μετασχηματίσουμε διαδοχικά το χρονοδιάγραμμα B στο χρονοδιάγραμμα O. Θα χρησιμοποιήσουμε για τούτο μία ακολουθία ανταλλαγών καθεμιά από τις οποίες ανταλλάσσει δύο εργασίες στο χρονοδιάγραμμα B οι οποίες εκτελούνται με διαφορετική σειρά στο χρονοδιάγραμμα O. Το σημαντικό είναι να δείξουμε ότι κάθε ανταλλαγή έχει ως αποτέλεσμα την μη αύξηση του σταθμισμένου αθροίσματος. Αυτό συνεπάγεται ότι το τελικό χρονοδιάγραμμα που θα προκύψει από τους μετασχηματισμούς μας δεν έχει μεγαλύτερο σταθμισμένο άθροισμα από το αρχικό χρονοδιάγραμμα B. Αφού το αρχικό χρονοδιάγραμμα B ήταν βέλτιστο, έπεται ότι και το τελικό χρονοδιάγραμμα θα είναι βέλτιστο. Αφού το τελικό αυτό χρονοδιάγραμμα ταυτίζεται με το άπληστο χρονοδιάγραμμα O, έπεται ότι το άπληστο χρονοδιάγραμμα O είναι βέλτιστο. Προχωράμε τώρα με τις τυπικές λεπτομέρειες της απόδειξής μας.

Η απόδειξη βελτιστότητας με επιχείρημα ανταλλαγής: Αφού τα χρονοδιαγράμματα O και B είναι διαφορετικά, υπάρχει ζεύγος εργασιών i και j τέτοιες ώστε η j προηγείται της i στο χρονοδιάγραμμα O, ενώ η i προηγείται της j στο χρονοδιάγραμμα B. Χωρίς βλάβη της γενικότητας, μπορούμε να υποθέσουμε ότι οι εργασίες i και j είναι διαδοχικές στο χρονοδιάγραμμα B. (Αν όλα τα ζεύγη διαδοχικών εργασιών στο χρονοδιάγραμμα O είχαν την ίδια σειρά και στο χρονοδιάγραμμα B, τότε τα χρονοδιαγράμματα O και B θα ήταν τα ίδια και το χρονοδιάγραμμα O θα ήταν βέλτιστο.) Από τον ορισμό του απλήστου αλγορίθμου, έχουμε ότι $\frac{w_j}{t_j} \geq \frac{w_i}{t_i}$.

Έστω C ο χρόνος ολοκλήρωσης της εργασίας που προηγείται της εργασίας i στο χρονοδιάγραμμα B. Τότε, η συνεισφορά των εργασιών i και j στο σταθμισμένο άθροισμα $\sum_{i=1}^{n} w_i C_i$ για το χρονοδιάγραμμα B είναι $\Delta_{ij} = w_i(C + t_i) + w_j(C + t_i + t_j)$.

Ανταλλάσσουμε τις εργασίες i και j στο χρονοδιάγραμμα B. Έστω $\overline{B}$ το χρονοδιάγραμμα που προκύπτει. Παρατηρούμε ότι οι χρόνοι ολοκλήρωσης όλων των εργασιών πλην των i και j παραμένουν οι ίδιοι στο χρονοδιάγραμμα $\overline{B}$ όπως ήταν στο χρονοδιάγραμμα B. Επομένως, μας ενδιαφέρει μόνο η συνεισφορά των εργασιών i και j στο σταθμισμένο άθροισμα $\sum_{i=1}^{n} w_i C_i$ για το χρονοδιάγραμμα $\overline{B}$, η οποία είναι είναι $\overline{\Delta}_{ij} = w_j(C + t_j) + w_i(C + t_j + t_i)$. Έχουμε ότι

$$\begin{aligned} & \Delta_{ij} - \overline{\Delta}_{ij} \\ = \;& w_j t_i - w_i t_j \\ \geq \;& 0 \qquad \text{(αφού } \frac{w_j}{t_j} \geq \frac{w_i}{t_i}\text{)}. \end{aligned}$$

Επομένως, το σταθμισμένο άθροισμα για το χρονοδιάγραμμα $\overline{B}$ δεν αυξάνει. Αφού το χρονοδιάγραμμα B ήταν βέλτιστο, έπεται ότι και το χρονοδιάγραμμα $\overline{B}$ είναι βέλτιστο. Αφού το τελικό χρονοδιάγραμμα $\overline{B}$ ταυτίζεται με το άπληστο χρονοδιάγραμμα O, έπεται ότι το άπληστο χρονοδιάγραμμα O είναι βέλτιστο.

Πολυπλοκότητα: Η πολυπλοκότητα του αλγόριθμου προέρχεται από την ταξινόμηση των λόγων $\frac{w_i}{t_i}$, όπου $1 \leq i \leq n$, και είναι $\Theta(n \lg n)$.