Λύσεις 5ης Σειράς Ασκήσεων

1. [Διασκευή του Προβλήματος 13.1 από το βιβλιο Algorithm Design των Kleinberg και Tardos.] Στο πρόβλημα του 3-Χρωματισμού, μας δίνεται ένας γράφος $G = (V, E)$ και θέλουμε να χρωματίσουμε κάθε κορυφή του με ένα από τρία χρώματα, ακόμη και αν δεν είναι δυνατό να δώσουμε διαφορετικά χρώματα σε κάθε ζευγάρι γειτονικών κορυφών. Λέμε ότι μία ακμή (u, v) ικανοποιείται αν τα χρώματα που αποδίδονται στις κορυφές u και v είναι διαφορετικά. Θέλουμε να μεγιστοποιήσουμε τον αριθμό των ακμών που ικανοποιούνται. Έστω c^* ο μέγιστος δυνατός αριθμός ακμών που ικανοποιούνται σε έναν 3-χρωματισμό.
Παρουσιάστε και αναλύστε έναν πιθανοτικό (πολυωνυμικό) αλγόριθμο ο οποίος παράγει έναν 3-χρωματισμό στον οποίο ο αναμενόμενος αριθμός m των ακμών που ικανοποιούνται θα είναι $m \geq \frac{2}{3}\, c^*$.

Λύση: Έχουμε τον εξής αλγόριθμο:

- Για κάθε κορυφή $v \in V$ (και ανεξάρτητα από τις υπόλοιπες):
  - Επίλεξε ένα από τα τρία χρώματα ομοιόμορφα και τυχαία.
    (Έτσι, το κάθε χρώμα επιλέγεται με πιθανότητα $\frac{1}{3}$.)
  - Χρωμάτισε την κορυφή v με το χρώμα που επέλεξες.

Για κάθε ακμή $e \in E$, ορίζουμε την τυχαία μεταβλητή X_e ως εξής:

$$X_e = \begin{cases} 1 & \text{αν η ακμή } e \text{ ικανοποιείται} \\ 0 & \text{αλλοιώς} \end{cases} .$$

Ορίζουμε την τυχαία μεταβλητή X ως τον αριθμό των ακμών που ικανοποιούνται. Προφανώς, $X = \sum_{e \in E} X_e$. Έχουμε να δείξουμε ότι $\mathcal{E}(X) \geq \frac{2}{3}\, c^*$.

Για οποιαδήποτε ακμή $e \in E$, υπάρχουν 9 τρόποι να χρωματίσουμε τις δύο κορυφές της. Σε 6 από τους 9 τρόπους, η ακμή e ικανοποιείται, ενώ η e δεν ικανοποιείται στους υπόοιπους τρόπους. Επομένως,

$$\mathbf{P}[\text{η ακμή } e \text{ ικανοποιείται}] = \frac{2}{3},$$

και

$$\begin{aligned} \mathcal{E}(X_e) &= 1 \cdot \mathbf{P}[\text{η ακμή } e \text{ ικανοποιείται}] + 0 \cdot \mathbf{P}[\text{η ακμή } e \text{ δεν ικανοποιείται}] \\ &= \frac{2}{3}. \end{aligned}$$

Έτσι, λαμβάνουμε ότι

$$\begin{aligned}\mathcal{E}(X) &= \mathcal{E}(\textstyle\sum_{e\in E} X_e) \\ &= \textstyle\sum_{e\in E} \mathcal{E}(X_e) \quad \text{(από την γραμμικότητα της αναμενόμενης τιμής)} \\ &= \frac{2}{3}|E| \\ &\geq \frac{2}{3}c^* \qquad \text{(αφού } c^* \leq |E|\text{)} \ ,\end{aligned}$$

όπως χρειάζεται.

2. [Διασκευή του Προβλήματος 13.12 από το βιβλίο Algorithm Design των Kleinberg και Tardos.] Θεωρείστε την ακόλουθη παραλλαγή του αλγορίθμου συρρίκνωσης για την εύρεση μίας ελάχιστης αποκοπής (η οποία ξεχωρίζει την πηγή s από τον προορισμό t) σε ένα δίκτυο ροής με μοναδιαίες χωρητικότητες.

- Συρρικνώνουμε τις ακμές επαναληπτικά ως εξής:
  - Σε μία δεδομένη επανάληψη, έστω s και t οι υπερκορυφές που περιέχουν τις αρχικές κορυφές s και t, αντίστοιχα.
  - Διαγράφουμε τις ακμές που συνδέουν τις κορυφές s και t και επιλέγουμε μία ακμή ομοιόμορφα και τυχαία μεταξύ εκείνων που απομένουν, την οποία και συρρικνώνουμε.

Δώστε ένα παράδειγμα για να αποδείξετε ότι η πιθανότητα να βρει ο αλγόριθμος αυτός μία ελάχιστη αποκοπή σε ένα δίκτυο ροής είναι εκθετικά μικρή.

Λύση: Θεωρούμε το δίκτυο ροής $G = (V, E)$ ως εξής:

- $V = \{s, t\} \cup \{v_1, \ldots, v_{n-2}\}$.
- Για κάθε κορυφή v_i, όπου $1 \leq i \leq n-2$, υπάρχουν 2 παράλληλες ακμές από την πηγή s προς την κορυφή v_i και μία ακμή από την κορυφή v_i προς τον προορισμό t. (Όλες οι ακμές έχουν χωρητικότητα 1.)

Προφανώς, η ελάχιστη αποκοπή είναι η αποκοπή $(\{s, v_1, \ldots, v_{n-2}\}, \{t\})$ με χωρητικότητα $n-2$.

Θεωρούμε οποιοδήποτε επαναληπτικό βήμα του αλγορίθμου στο οποίο επιλέγεται (ομοιόμορφα και τυχαία) κάποια ακμή. Υπάρχουν δύο περιπτώσεις:

- Αν επιλέξουμε κάποια ακμή (s, v_i), όπου $1 \leq i \leq n-2$, τότε η κορυφή v_i θα συρρικνωθεί με την πηγή s (και οι δύο ακμές (s, v_i) θα διαγραφούν). Ως αποτέλεσμα, πριν την επόμενη επιλογή ακμής, θα διαγράψουμε την ακμή (v_i, t).
- Αν επιλέξουμε κάποια ακμή (v_i, t), όπου $1 \leq i \leq n-2$, τότε η κορυφή v_i θα συρρικνωθεί με την πηγή t (και η ακμή (v_i, t) θα διαγραφεί). Ως αποτέλεσμα, πριν την επόμενη επιλογή ακμής, θα διαγράψουμε τις δύο παράλληλες ακμές (s, v_i).

Επομένως, κάθε επαναληπτικό βήμα του αλγόριθμου έχει σαν αποτέλεσμα την διαγραφή ενός διπλού μονοπατιού μήκους 2 από την πηγή s προς τον προορισμό t, και τη συρρίκνωση κάποιας κορυφής v_i, όπου $1 \leq i \leq n-2$, με την πηγή s ή τον προορισμό t. Συνολικά, λοιπόν, υπάρχουν $n-2$ επαναλήψεις. Αφού υπάρχουν δύο παράλληλες ακμές (s, v_i) και μία μόνο ακμή (v_i, t) για κάθε κορυφή v_i, όπου $1 \leq i \leq n-2$, έπεται ότι για κάθε δεδομένη επανάληψη, η πιθανότητα να συρρικνωθεί κάποια κορυφή v_i, όπου $1 \leq i \leq n-2$, με την πηγή s είναι ίση με $\frac{2}{3}$, ενώ η πιθανότητα να συρρικνωθεί κάποια κορυφή v_i, όπου $1 \leq i \leq n-2$, με τον προορισμό t είναι ίση με $\frac{1}{3}$,

Προφανώς, ο αλγόριθμος επιστρέφει την ελάχιστη αποκοπή αν και μόνο αν σε καμία επανάληψη δεν γίνεται συρρίκνωση κάποιας κορυφής v_i με τον προορισμό t. Αφού οι επαναλήψεις είναι ανεξάρτητες, έπεται ότι η πιθανότητα ώστε ο αλγόριθμος να επιστρέψει την ελάχιστη αποκοπή είναι ίση με $\left(\frac{2}{3}\right)^{n-2}$, δηλαδή είναι εκθετικά μικρή.

3. [Διασκευή του Προβλήματος 13.17 από το βιβλιο Algorithm Design των Kleinberg και Tardos.] Παίζουμε ένα τυχερό παιγνίδι στο οποίο κάνουμε μία σειρά από n παρτίδες. Σε κάθε παρτίδα, το κέρδος αυξάνεται κατά 1 με πιθανότητα $\frac{1}{3}$ και μειώνεται κατά 1 με πιθανότητα $\frac{2}{3}$. Αρχικά, το κέρδος είναι 0.
Δείξτε ότι ο αναμενόμενος αριθμός παρτίδων μετά από τις οποίες το κέρδος είναι θετικό είναι φραγμένο από μία σταθερά c (η οποία είναι ανεξάρτητη του n).

Λύση: Παραλείπεται.

4. [Διασκευή του Προβλήματος 10.9 από το βιβλιο Algorithm Design των Kleinberg και Tardos.] Μας δίνεται ένα δυαδικό δένδρο $T = (V, E)$ με άρτιο αριθμό κορυφών, και ένα μη αρνητικό βάρος σε κάθε ακμή. Θέλουμε να βρούμε μία διαμέριση του συνόλου κορυφών V σε δύο σύνολα V_1 και V_2 ίσου μεγέθους έτσι η βαρύτητα της αποκοπής μεταξύ των δύο συνόλων να μεγιστοποιηθεί. (Η βαρύτητα της αποκοπής είναι το συνολικό βάρος των χιαστί ακμών ως προς την αποκοπή.)
Παρουσιάστε και αναλύστε ένα πολυωνυμικό αλγόριθμο για αυτό το πρόβλημα.

Λύση: Θα χρησιμοποιήσουμε την μέθοδο του δυναμικού προγραμματισμού. Για οποιαδήποτε ακμή (u, v) στο δυαδικό δένδρο $\mathsf{T} = (\mathsf{V}, \mathsf{E})$, συμβολίζουμε ως $\mathsf{w}(u, v)$ το βάρος της ακμής. Κατονομάζουμε αυθαίρετη κορυφή r ως τη ρίζα του δυαδικού δένδρου T. Αυτό επάγει, για κάθε κορυφή $v \in V$, ένα υποδένδρο T_v με ρίζα την κορυφή v. Συμβολίζουμε ως $\mathsf{V}(T_v)$ το σύνολο κορυφών στο υποδένδρο T_v, με $n_v = |T_v|$.

Τα υποπροβλήματα. Για κάθε κορυφή $v \in V$ και για κάθε ακέραιο k όπου $0 \leq k \leq n_v$, ορίζουμε το υπολογιστικό πρόβλημα $\Pi(v, k)$ ως εξής:

> Με δεδομένο το δυαδικό δένδρο T_v, βρες μία διαμέριση του συνόλου των κορυφών του σε δύο σύνολα, όπου το ένα από τα δύο σύνολα το οποίο περιέχει την κορυφή v έχει μέγεθος k, έτσι ώστε η βαρύτητα της αποκοπής μεταξύ των δύο συνόλων να μεγιστοποιηθεί.

Προφανώς, υπάρχουν το πολύ $|\mathsf{V}| \cdot (|\mathsf{V}|+1) = O(|\mathsf{V}|^2)$ τέτοια υποπροβλήματα, και το υπολογιστικό πρόβλημα που έχουμε να επιλύσουμε είναι το υποπρόβλημα $\Pi\left(r, \frac{|V|}{2}\right)$.

<u>Η δομή της βέλτιστης λύσης.</u> Θα αποδείξουμε την εξής πρόταση:

Πρόταση 1. Έστω μία βέλτιστη λύση (V_1, V_2) για το (υπο)πρόβλημα $\Pi\left(r, \frac{|V|}{2}\right)$. Θεωρούμε αυθαίρετη κορυφή $v \in V_1$ και υποθέτουμε ότι $|V_1 \cap \mathsf{V}(T_v)| = k$. Τότε, η επαγόμενη διαμέριση $(V_1 \cap \mathsf{V}(T_v), V_2 \cap \mathsf{V}(T_v))$ είναι μία βέλτιστη λύση για το υποπρόβλημα $\Pi(v, k)$.

Η απόδειξη θα γίνει διά της εις άτοπον απαγωγής. Θα κατασκευάσουμε από μία λύση καλύτερη από την διαμέριση $(V_1 \cap \mathsf{V}(T_v), V_2 \cap \mathsf{V}(T_v))$ για το υποπρόβλημα $\Pi(v, k)$ μία λύση καλύτερη από την βέλτιστη λύση (V_1, V_2) για το πρόβλημα $\Pi\left(r, \frac{|V|}{2}\right)$.

<u>**Απόδειξη:**</u> Υποθέτουμε, για να φτάσουμε σε αντίφαση, ότι υπάρχει διαμέριση (V_1', V_2') για το σύνολο κορυφών $\mathsf{V}(\mathsf{T}_v)$ με βαρύτητα αποκοπής <u>μεγαλύτερη</u> από εκείνη για την διαμέριση $(V_1 \cap \mathsf{V}(T_v), V_2 \cap \mathsf{V}(T_v))$. Χρησιμοποιούμε την διαμέριση (V_1', V_2') (για το σύνολο κορυφών $\mathsf{V}(\mathsf{T}_v)$) για να κατασκευάσουμε μία διαμέριση για το σύνολο κορυφών V ως εξής:

> Αντικαθιστούμε το υποσύνολο $V_1 \cap \mathsf{V}(T_v)$ στο σύνολο κορυφών V_1 με το σύνολο κορυφών V_1', και αντικαθιστούμε το σύνολο κορυφών $V_2 \cap \mathsf{V}(T_v)$ στο σύνολο κορυφών V_2 με το σύνολο κορυφών V_2'.

Αφού το T είναι δένδρο, η βαρύτητα της αποκοπής που προκύπτει μείον την βαρύτητα της αρχικής αποκοπής ισούται με το συνολικό βάρος των χιαστί ακμών ως προς την διαμέριση (V_1', V_2') (για το σύνολο κορυφών $\mathsf{V}(\mathsf{T}_v)$) μείον το συνολικό βάρος των χιαστί ακμών ως προς την διαμέριση $(V_1 \cap \mathsf{V}(T_v), V_2 \cap \mathsf{V}(T_v))$ (για το σύνολο κορυφών $\mathsf{V}(\mathsf{T}_v)$). Από την υπόθεσή μας, έπεται ότι η βαρύτητα της αποκοπής που προκύπτει είναι μεγαλύτερη από την βαρύτητα της αρχικής αποκοπής (V_1, V_2). Αντίφαση. □

<u>Η τιμή της βέλτιστης λύσης.</u> Έστω $\mathsf{W}(v, k)$ η <u>τιμή</u> της βέλτιστης λύσης για το υποπρόβλημα $\Pi(v, k)$ — δηλαδή, $\mathsf{W}(v, k)$ είναι η <u>μέγιστη δυνατή</u> βαρύτητα αποκοπής μεταξύ δύο συνόλων κορυφών V_1 και V_2 τα οποία διαμερίζουν το σύνολο κορυφών T_v έτσι ώστε $v \in V_1$ με $|V_1| = k$. Θα αποδείξουμε μία αναδρομική σχέση για την τιμή $\mathsf{W}(v, k)$, όπου $v \in \mathsf{V}$ και $k \leq |\mathsf{V}(\mathsf{T}_v)|$. Αν $k = 0$, θέτουμε αυθαίρετα για κάθε κορυφή $v \in \mathsf{V}$, $\mathsf{W}(v, 0) := 0$. Υποθέτουμε λοιπόν ότι $k \geq 1$. Διακρίνουμε δύο περιπτώσεις.

- Έστω ότι η κορυφή v είναι φύλλο. Τότε, $|\mathsf{V}(\mathsf{T}_v)| = 1$, οπότε $k \leq 1$. Έπεται ότι έχουμε ένα μοναδικό υποπρόβλημα $\Pi(v, 1)$, και προφανώς $\mathsf{W}(v, 1) = 0$.
- Έστω ότι η κορυφή v δεν είναι φύλλο. Τότε, το σύνολο V_1 περιέχει την κορυφή v και άλλες $k-1$ κορυφές. Υπάρχουν δύο υποπεριπτώσεις:
  - Η κορυφή v έχει ένα μοναδικό παιδί u. Τότε, όλες οι υπόλοιπες $k-1$ κορυφές (στο σύνολο V_1) περιέχονται στο υποδένδρο T_u. Έτσι, $|\mathsf{V}_1 \cap \mathsf{V}(\mathsf{T}_u)| = k-1$ και $|\mathsf{V}_2 \cap \mathsf{V}(\mathsf{T}_u)| = n_u - (k-1)$. Υπάρχουν και πάλι δύο υποπεριπτώσεις.

* Το παιδί $u \in V_1$. Τότε, $u \in \mathsf{V}_1 \cap \mathsf{V}(\mathsf{T}_u)$. Η Πρόταση 1 συνεπάγεται ότι μία βέλτιστη λύση για το υποπρόβλημα $\Pi(v, k)$ επάγει μία βέλτιστη λύση για το υποπρόβλημα $\Pi(u, k-1)$. Έτσι, $\mathsf{W}(v, k) = \mathsf{W}(u, k-1)$.
* Το παιδί $u \in V_2$. Τότε, $u \in \mathsf{V}_2 \cap \mathsf{V}(\mathsf{T}_u)$ και η ακμή (u, v) είναι χιαστί ακμή (για την διαμέριση $(\mathsf{V}_1, \mathsf{V}_2)$). Η Πρόταση 1 συνεπάγεται ότι μία βέλτιστη λύση για το υποπρόβλημα $\Pi(v, k)$ επάγει μία βέλτιστη λύση για το υποπρόβλημα $\Pi(u, n_u - (k-1))$. Έτσι, $\mathsf{W}(v, k) = \mathsf{w}(u, v) + \mathsf{W}(u, n_u - (k-1))$.

Συνολικά λοιπόν έχουμε ότι

$$\mathsf{W}(v, k) \quad = \quad \max\{\mathsf{W}(u, k-1), \mathsf{w}(u, v) + \mathsf{W}(u, n_u - (k-1))\}\,.$$

– Η κορυφή v έχει δύο παιδιά u και w. Τότε, οι υπόλοιπες $k-1$ κορυφές (στο σύνολο V_1) περιέχονται στα υποδένδρα T_u και T_v. Έτσι, $|\mathsf{V}_1 \cap (\mathsf{V}(\mathsf{T}_u)\mathsf{V}(\mathsf{T}_w))| = k-1$ και $|\mathsf{V}_2 \cap (\mathsf{V}(\mathsf{T}_u)\mathsf{V}(\mathsf{T}_w))| = (n_u + n_w) - (k-1)$.
Υποθέτουμε ότι $|\mathsf{V}_1 \cap \mathsf{V}(\mathsf{T}_u)| = k_1$ και $|\mathsf{V}_1 \cap \mathsf{V}(\mathsf{T}_w)| = k_2$, όπου $k_1 + k_2 = k-1$. Έπεται ότι $|\mathsf{V}_2 \cap \mathsf{V}(\mathsf{T}_u)| = n_u - k_1$ και $|\mathsf{V}_2 \cap \mathsf{V}(\mathsf{T}_w)| = n_w - k_2$. Για το παιδί u (αντίστοιχα, το παιδί w), υπάρχουν δύο υποπεριπτώσεις: $u \in \mathsf{V}_1$ και $u \in \mathsf{V}_2$ (αντίστοιχα, $w \in \mathsf{V}_1$ και $w \in \mathsf{V}_2$). Επειδή το T είναι δένδρο, οι υποπεριπτώσεις μπορούν να συνδυαστούν ανεξάρτητα. Έτσι, η ανάλυση για την περίπτωση όπου η κορυφή v είχε μόνο ένα παιδί συνεπάγεται ότι

$$\mathsf{W}(v, k) \quad = \quad \max_{0 \le k_1 \le k-1, k_2 = k-1-k_2} \left(\begin{array}{l} \max\{\mathsf{W}(u, k_1), \mathsf{w}(u, v) + \mathsf{W}(u, n_u - k_1)\} \\ + \\ \max\{\mathsf{W}(w, k_2), \mathsf{w}(w, v) + \mathsf{W}(w, n_w - k_2)\} \end{array} \right).$$

Ο αλγόριθμος δυναμικού προγραμματισμού. Παρουσιάζουμε τώρα τον αλγόριθμό μας:

- Συμπλήρωση του πίνακα με τις λύσεις στα υποπροβλήματα: Θα χρησιμοποιήσουμε ένα πίνακα $\mathsf{V} \times \{0, 1, |V|\}$ για να καταχωρήσουμε όλες τις τιμές $\mathsf{W}(v, k)$, όπου $v \in \mathsf{V}$ και $0 \leq k \leq |\mathsf{V}|$, ως εξής:

  - Αν $k = 0$, θέτουμε για κάθε κορυφή $v \in \mathsf{V}$, $\mathsf{W}(v, 0) := 0$.
  - Αλλοιώς ($k \geq 1$), χρησιμοποιούμε την τιμή της βέλτιστης λύσης για να υπολογίσουμε όλες τις τιμές (ξεκινώντας από τα φύλλα και προχωρώντας προς μεγαλύτερα υποδένδρα).

    Κατά την εξέταση της κορυφής $v \in \mathsf{V}$ η οποία δεν είναι φύλλο, καταχωρούμε για κάθε ακέραιο k όπου $0 \leq k \leq |\mathsf{V}(\mathsf{T}_v)|$ τις αποφάσεις μας να συμπεριλάβουμε το παιδί ή τα παιδιά της κορυφής v στο συνολο κορυφών V_1 ως εξής:

    * Έστω ότι η κορυφή v έχει μόνο ένα παιδί u. Τότε, για οποιοδήποτε ακέραιο k όπου $0 \leq k \leq |\mathsf{V}(\mathsf{T}_v)|$, συμπεριλαμβάνουμε την κορυφή u στο σύνολο κορυφών V_1 αν και μόνο αν $\mathsf{W}(u, k-1) \geq \mathsf{w}(u, v) + \mathsf{W}(u, n_u - (k-1))$.
    * Έστω ότι η κορυφή v έχει δύο παιδιά u kai w. Έστω κ_1 και κ_2 (με $\kappa_1 + \kappa_2 = k$ οι δύο ακέραιοι που προσδιορίζουν την τιμή $\mathsf{W}(v, k)$ (σύμφωνα με την τιμή της βέλτιστης λύσης). Τότε, συμπεριλαμβάνουμε την κορυφή u στο σύνολο κορυφών V_1 αν και μόνο αν $\mathsf{W}(u, k_1) \geq \mathsf{w}(u, v) + \mathsf{W}(u, n_u - k_1$. Τέλος, συμπεριλαμβάνουμε την κορυφή w στο σύνολο κορυφών V_1 αν και μόνο αν $\mathsf{W}(w, k_2) \geq \mathsf{w}(w, v) + \mathsf{W}(w, n_w - k_2$.

    Συμπεριλαμβάνουμε την ρίζα r στο σύνολο V_1.

- Ανακατασκευή της βέλτιστης λύσης: Η τιμή της βέλτιστης λύσης που ζητούμε είναι η τιμή $\mathsf{W}\left(r, \frac{|\mathsf{V}|}{2}\right)$ που έχουμε στη θέση $\left(r, \frac{|\mathsf{V}|}{2}\right)$ του πίνακά μας. Οι καταχωρημένες αποφάσεις μας (να συμπεριλάβουμε ή όχι κάποια κορυφή στο σύνολο κορυφών V_1) κατασκευάζουν αυτόματα την βέλτιστη λύση.

$\boxed{\text{Πολυπλοκότητα.}}$ Χρειάζονται $O(|\mathsf{V}|)$ βήματα για τον υπολογισμό της τιμής σε οποιαδήποτε θέση του πίνακα με τις λύσιες στα υποπροβλήματα. Έτσι, χρειάζονται συνολικά $O(|\mathsf{V}|^3)$ βήματα για τη συμπλήρωση του πίνακα με τις λύσιες στα υποπροβλήματα. Αφού η ανακατασκευή της βέλτιστης λύσης είναι αυτόματη, αυτός είναι και ο συνολικός αριθμός βημάτων.