תרגיל 1 – עפ"מ צהוב ביותר

יהא $G=(V,E)$ גרף לא מכוון וקשיר עם משקלות על הקשתות $w:E\to N$. נניח גם כי כל קשת צבועה או בשחור או בצהוב. הצע אלגוריתם המוצא עפ"מ צהוב ביותר ב-G, דהיינו עפ"מ עם מספר גדול ביותר אפשרי של קשתות צהובות.

רעיון

נרצה לתת לקשתות צהובות עדיפות על פני קשתות שחורות מאותו המשקל. לשם כך נגדיר פונקצית משקל חדשה

$$w'(e)=\begin{cases} w(e)-1/n^2 & e \text{ is yellow} \\ w(e) & \text{otherwise} \end{cases}$$

נסמן ב-$y(T)$ את מספר הקשתות הצהובות ב-T. נבחין כי

$$w'(T)=w(T)-y(T)/n^2$$

טענה 1

T הוא עפ"מ צהוב ביותר לפי פונקצית המשקל w אם"ם T הוא עפ"מ לפי פונקצית המשקל w'.

הוכחה

ראשית ניקח שני עצים T_1,T_2 כך ש-$w(T_1)>w(T_2)$, אז

$$w'(T_1)-w'(T_2)=\underbrace{w(T_1)-w(T_2)}_{\geq 1}-\frac{1}{n^2}\left(\underbrace{y(T_2)-y(T_1)}_{\leq\binom{n}{2}}\right)>0$$

על כן

$$w(T_1) > w(T_2) \quad \Rightarrow \quad w'(T_1) > w'(T_2)$$

כלומר, המונוטוניות של משקלי העצים נשמרת וזאת ללא קשר לצבע הקשתות, ובפרט העצים היחידים שעלולים להיות עפ"מים לפי w' הם העפ"מים לפי w (במילים אחרות, עץ שלא היה עפ"מ לפי w לא יכול להיות עפ"מ לפי w' רק בגלל שיש לו הרבה קשתות צהובות, פשוט כי למרות עדיפותן על קשתות שחורות, השפעתן לא מספיקה (וזאת בגלל הפקטור הכפלי $1/n^2$ שנתנו להן)).

שנית, נבחין כי אם $w(T_1) = w(T_2)$ אז $w'(T_1) < w'(T_2)$ אם"ם $y(T_1) > y(T_2)$.

מכאן שקבוצת העפ"מים של G לפי w' היא תת הקבוצה של העפ"מים של G לפי w עם מספר מקסימאלי של קשתות צהובות.

■

אלגוריתם

<u>קלט</u>: גרף לא מכוון וקשיר $G = (V, E)$ ופונקצית משקל $w: E \to N$.

<u>פלט</u>: עפ"מ צהוב ביותר של G.

1. חשב את פונקצית המשקל w'.
2. מצא עפ"מ T לפי פונקצית המשקל w', והחזר את T כפלט.

נכונות

נכונות האלגוריתם נובעת מטענה 1.

סיבוכיות

מכיוון שחישוב w' ניתן להעשות בזמן לינארי באורך הקלט, סיבוכיות האלגוריתם היא כסיבוכיות האלגוריתם אותו אנו מפעילים בשלב 2 למציאת עפ"מ. שני האלגוריתמים שראינו בקורס רצים בזמן $O(E \log V)$ ולכן זוהי גם סיבוכיות זמן הריצה של האלגוריתם אותו אנו מנתחים.

תרגיל 1

יהא $G=(V,E)$ גרף לא מכוון וקשיר עם פונקצית משקל $w:E\to R$.

1. הצע אלגוריתם אשר בהינתן קשת $e\in E$ מכריע האם קיים עפ"מ של G המכיל את e.
2. הצע אלגוריתם אשר בהינתן קשת $e\in E$ מכריע האם כל עפ"מ של G מכיל את e.

פתרון

ראשית נתאר אלגוריתמים לשתי הבעיות, הרצים בזמן $O(E\log V)$. לפתרון הסעיף הראשון נציע את האלגוריתם הבא:

אלגוריתם (לא מספיק יעיל) לפתרון סעיף 1

קלט: גרף $G=(V,E)$ לא מכוון וקשיר עם פונקצית משקל $w:E\to R$, וקשת $e\in E$.

פלט: הכרעה בדבר האם קיים עפ"מ של G המכיל את e.

1. צבע את e בצהוב ואת קבוצת הקשתות $E\setminus\{e\}$ בשחור.
2. הרץ את האלגוריתם למציאת עפ"מ צהוב ביותר. נסמן את העץ המתקבל ב-T.
3. החזר "כן" אם ורק אם e היא קשת ב-T.

נכונות

אם אין עפ"מ המכיל את e אז מנכונות קרוסקל (המורץ על ידי האלגוריתם למציאת עפ"מ צהוב ביותר (אפשר גם פרים)), בשלב (2) T המתקבל לא יכיל e ולכן בשלב (3) האלגוריתם יחזיר "לא". מאידך, נבחין כי קבוצת העפ"מים המכילים את e הם בדיוק קבוצת העפ"מים הצהובים ביותר. לכן אם יש עפ"מ המכיל את e אז הוא יתגלה על ידי האלגוריתם משלב (2) וזאת מנכונות האלגוריתם למציאת עפ"מ צהוב ביותר, ולכן בשלב (3) האלגוריתם יחזיר "כן".

סיבוכיות

שלב (1) ושלב (3) יכולים להתבצע בזמן לינארי. שלב (2) עולה $O(E\log V)$.

אלגוריתם (לא מספיק יעיל) לפתרון בעיה (2)

1. צבע את e בשחור ואת קבוצת הקשתות $E\setminus\{e\}$ בצהוב.
2. הרץ את האלגוריתם למציאת עפ"מ צהוב ביותר. נסמן את העץ המתקבל ב-T.
3. החזר "כן" אם ורק אם e היא קשת ב-T.

הוכחת נכונות

אם כל עפ"מ מכיל את e אז מנכונות קרוסקל (המורץ על ידי האלגוריתם למציאת עפ"מ צהוב ביותר), T המתקבל בשלב (2) יכיל את e ולכן האלגוריתם יחזיר "כן" בשלב (3). מצד שני, קבוצת העפ"מים שאינם מכילים את e היא בדיוק קבוצת העפ"מים הצהובים ביותר. לכן, אם קיים עפ"מ שאינו מכיל את e, בסיום שלב (2) T לא יכיל את e, וזאת מנכונות האלגוריתם למציאת עפ"מ צהוב ביותר, ולכן בשלב (3) האלגוריתם יחזיר "לא".

סיבוכיות

כמו באלגוריתם (הלא מספיק יעיל) לפתרון סעיף 1.

לסעיף 2 ניתן למצוא אלגוריתם אפילו פשוט יותר שרץ ב- $O(E\log V)$:

אלגוריתם (לא מספיק יעיל נוסף) לפתרון בעיה (2)

1. חשב את משקלו של עפ"מ של G.
2. הסר את e וחשב את משקלו של עפ"מ בגרף החדש (אם בכלל קיים).
3. אם לא קיים עפ"מ בגרף החדש, או שמשקלו של העפ"מ בגרף החדש שווה למשקלו של העפ"מ בגרף המקורי, החזר "כן". אחרת החזר "לא".

מסתבר שניתן לפתור את תרגיל 1, על שני סעיפיו, בזמן לינארי (ובפרט, בכדי לפתור את הבעיות, לא צריך כלל לחשב עפ"מ). לצורך כך נציג כעת זוג טענות המספקות אפיונים לכך שקשת נמצאת באיזשהו עפ"מ ולכך שקשת נמצאת בכל עפ"מ. אפיונים אשר קל יותר לוודא אותם מבחינה חישובית.

טענה 1

קשת e נמצאת באיזשהו עפ"מ של G אם"ם כל מעגל המכיל את e מכיל קשת $e'\neq e$ כך ש- $w(e')\geq w(e)$.

טענה 2

קשת e נמצאת בכל עפ"מ של G אם"ם כל מעגל המכיל את e מכיל קשת e' כך ש- $w(e')>w(e)$.

הוכחת טענה 1

<u>בכיוון אחד</u>: נסמן $e=uv$, ונניח כי קשת זו נמצאת בעפ"מ של G שנסמנו ב-T. נניח בשלילה שקיים מעגל C המכיל את e ושכל שאר קשתותיו הן ממשקל נמוך ממש מ-$w(e)$. הסרת e מ-T נותנת בידינו שני רכיבי קשירות שנסמנם A_u,A_v כך ש-$u\in A_u,v\in A_v$. נביט במסלול P המחבר בין u,v המתקבל על ידי הסרת e מ-C. תהא e' ב-P קשת החוצה את החתך (A_u,A_v). כפי שראינו בהוכחת הנכונות של האלגוריתם הגנרי למציאת עפ"מ בהרצאה, אם נוסיף את e' לשני רכיבי הקשירות נקבל עץ פורש של G. מכיוון ש-$w(e')<w(e)$ קיבלנו עץ פורש של G שהוא קל יותר מ-T בסתירה להיות T עפ"מ של G.

<u>בכיוון השני</u>: נניח שכל מעגל המכיל את e מכיל קשת $e'\neq e$ כך ש-$w(e')\geq w(e)$. יהא T עפ"מ כלשהו של G. אם רצה המקרה ו-T מכיל את e - סיימנו. אחרת נראה כיצד לבנות מ-T עפ"מ T' המכיל את e. יהא P המסלול (היחיד) המחבר בין u,v ב-T. מכיוון שהוספת e ל-P סוגרת מעגל המכיל את e ב-G, קיימת, מההנחה, קשת $e'\neq e$ על המעגל כך ש-$w(e')\geq w(e)$, כלומר קיימת קשת e' במסלול P כך ש-$w(e')\geq w(e)$. אם נסיר את e' מ-T ונוסיף לגרף המתקבל את e נקבל עפ"מ המכיל את e, וזאת שוב מאותם טיעונים שראינו בהוכחת הנכונות של האלגוריתם הגנרי למציאת עפ"מ בהרצאה.

■

הוכחת טענה 2

הוכחת טענה 2 דומה להוכחת טענה 1, ונשאיר אותה כתרגיל.

לכאורה נדמה שזוג הטענות נותנת אפיונים שהם קשים יותר לוידוא מבחינה חישובית, שכן אנו צריכים לעבור על כל המעגלים המכילים קשת בכדי להכריע האם היא נמצאת באיזשהו עפ"מ או לחילופין בכל עפ"מ. למרות שמשימה זו נשמעת מורכבת מבחינה חישובית (שכן ייתכנו מספר אקספוננציאלי של מעגלים בגרף אשר מכילים את הקשת), היא פשוטה למדי. נבחין כי כל מעגל המכיל את $e=uv$ מכיל קשת $e'\neq e$ כך ש-$w(e')\geq w(e)$ אם ורק אם בגרף המתקבל מ-G על ידי הסרת כל הקשתות עם משקל לפחות $w(e)$ אין מסלול מ-u ל-v. באופן דומה, כל מעגל המכיל את e מכיל קשת e' כך ש-$w(e')>w(e)$ אם ורק אם בגרף המתקבל מ-G תוך הסרת e יחד עם כל הקשתות עם משקל הגדול ממש מ-$w(e)$ אין מסלול מ-u ל-v. אבחנה זו יחד עם זוג הטענות נותנת בידינו אלגוריתמים לזוג הבעיות.

אלגוריתם (יעיל) לפתרון סעיף 1

<u>קלט:</u> גרף $G=(V,E)$ לא מכוון וקשיר עם פונקצית משקל $w:E\to R$, וקשת $e\in E$.

<u>פלט:</u> הכרעה בדבר האם קיים עפ"מ של G המכיל את e.

1. בנה את הגרף $G'=(V,E')$ כך ש-$E'=\{e'\in E:w(e')<w(e)\}$.
2. החזר "כן" אם"ם לא קיים מסלול המחבר בין u,v ב-G' (את קיומו של מסלול כזה ניתן להכריע תוך הרצת BFS או DFS).

אלגוריתם (יעיל) לפתרון בעיה (2)

<u>קלט:</u> גרף $G=(V,E)$ לא מכוון וקשיר עם פונקצית משקל $w:E\to R$, וקשת $e\in E$.

<u>פלט:</u> הכרעה בדבר האם כל עפ"מ של G המכיל את e.

1. בנה את הגרף $G'=(V,E')$ כך ש-$E'=\{e'\in E:w(e')\le w(e)\}\setminus\{e\}$.
2. החזר "כן" אם"ם לא קיים מסלול המחבר בין u,v ב-G' (את קיומו של מסלול כזה ניתן להכריע תוך הרצת BFS או DFS).

סיבוכיות

בניית הגרף בשלב (1) בכל אחד מהאלגוריתמים עולה זמן לינארי. בדיקת הקשירות בשלב (2) בכל אחד מהאלגוריתמים עולה גם היא זמן לינארי (ליתר דיוק שלב זה עולה $O(V+E')$ אבל מכיוון ש-$|E'|\le|E|$ אז סיבוכיות שלב זה לינארית באורך הקלט). סך הכל לכן סיבוכיות האלגוריתמים היא לינארית.

.3

יהא G גרף קשיר ו- $W: E \to (0, \infty)$ פונקציית משקל על קשתותיו. תארו אלגוריתם למציאת עץ פורש G, שמכפלת משקליו מקסימלי.

<u>פתרון</u>

נבנה גרף חדש שבו $G1 = (V, E)$ עם פונקציית המשקל $\forall e \in E \quad w'(e) = \log(w(e))$.

נניח שהקשתות בעץ הפורש המבוקש הן: $e_1, e_2, e_3, ...e_n$.

לכן:

$$\max\{\prod_{i=1}^{n} w(e_i)\} = \max(\log(\prod_{i=1}^{n} w(e_i))) =$$

$$= \max(\log(w(e_1)) + \log(w(e_2)) + \log(w(e_3)) + ...\log(w(e_n)))$$

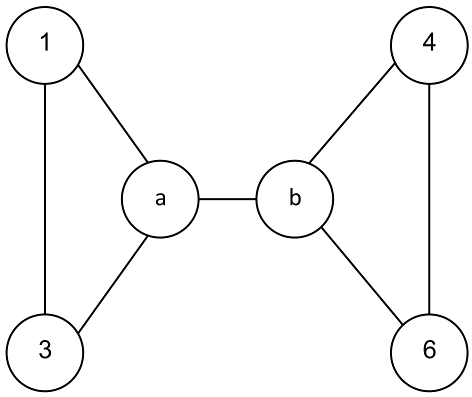
לכן נמצא עץ פורש מקסימלי ב- G1 ץ עץ זה הוא העץ הפורש המבוקש ב- G .

סיבוכיות: $O(E + V \log V)$.

4. הוכח או הפרך את הטענה הבאה : אם בגרף פשוט כלשהו G דרגתו של כל קודקוד היא לפחות 2 אזי בגרף G אין קשת מפרידה.

<u>פתרון:</u>

הפרכה:



5.

נתון גרף לא מכוון וקשיר G=(V,E) ופונקציית משקל ממשית של קשתותיו. כל קשת בגרף צבועה באחד משלושת הצבעים כחול, לבן או אדום.

תארו אלגוריתם יעיל, המוצא מבין כל העצים הפורשים שעבורם 2k + n – m מקסימלי, עץ שמשקלו מינימלי, כאשר k – מספר הקשתות הכחולות בעץ, n – מספר הקשתות הלבנות בעץ ו- m – מספר הקשתות האדומות בעץ.

מהי סיבוכיות זמן הריצה של האלגוריתם ? הוכיחו נכונותו.

פתרון

ניתן לכל קשת משקל נוסף:

לכחולות משקל (-2) , ללבנות משקל (-1) ולאדומות משקל 1.

לכל עץ פורש שעבורו 2k+n-m מקסימלי

מתקיים {-2k-n+m} מינימלית.

לכן הבעיה הנתונה שקולה למציאת עץ פורש מינימלי לפי פונקציית המשקל הנוספת שהגדרנו.

.6

נתון גרף G=(V,E) לא מכוון , קשיר ועם פונקציית משקל הבאה: $W:E\rightarrow\{1,3,10\}$ לכל קשת $e\in E$, כאשר הגרף מיוצג על ידי רשימות שכנות.

א. כתוב אלגוריתם יעיל המוצא קבוצת קשתות המכילה לפחות קשת אחת מכל מעגל בגרף ומשקלה הכולל מינימלי.

ב. מהי סיבוכיות זמן הריצה של האלגוריתם שהצעת ?

<u>**פתרון**</u>

<u>**האלגוריתם:**</u>

<u>צעד 1</u> : נריץ את האלגוריתם ___(1)______

<u>צעד 2</u> : קבוצת הקשתות המבוקשת הינה : _____(2)_____

באלגוריתם הנ"ל חסרים **שני** ביטויים המסומנים במספרים בין סוגריים עגולים. השלם אותם!

(1)- למציאת עץ פורש מקסימלי לפי קרוסקל.

(2) E-T

(3) סבוכיות זמן הריצה של האלגוריתם- כסיבוכיות זמן הריצה של קרוסקל .

<u>הערה חשובה כתרגיל לסטודנטים</u>

נניח שכל המשקולות בגרף הם מספרים שלמים בין 1 ל- W ,
עבור קבוע W כלשהו.

הראה כי זמן הריצה של פרים הוא:

$$O(V\log W + E + W) \simeq O(V\log W + E)$$