
<u>שאלה 1</u> (מבחינה של פרופ' נוגה אלון)
נתון גרף לא מכוון וקשיר $G(V,E)$ עם משקלים על הקשתות, ונתון עץ פורש מינימלי של G. נניח שמוסיפים ל- G קודקוד חדש v וקשתות ממושקלות המחברות אותו לחלק מקודקודי הגרף. תארו אלגוריתם שרץ בזמן $O(|V|\log|V|)$ ומוצא עץ פורש מינימלי בגרף החדש.

<u>תאור האלגוריתם</u>
נריץ את האלגוריתם של קרוסקל למציאת עץ פורש מינימלי על הגרף שקבוצת צמתיו היא קבוצת כל הצמתים כולל הצומת הנוסף וקבוצת קשתותיו היא קשתות העץ המקורי ובנוסף כל הקשתות שמחברות את הצומת הנוסף עם הצמתים האחרים.

<u>הוכחת נכונות</u>
<u>טענה</u>: קיים עץ פורש מינימלי שבו אין אף קשת מהגרף המקורי שלא היתה בעץ הפורש המקורי.
נניח שבעץ פורש מינימלי כן קיימת קשת e_1 כזאת. נראה שניתן להחליף אותה על-ידי קשת שכן היתה חלק מהעץ המקורי, זאת בלי להגדיל את משקל העץ. אם נסיר את e_1 מהעץ אז יתקבלו שני רכיבי קשירות. בעץ המקורי קיימת לפחות קשת אחת שיכולה לחבר את שני רכיבי הקשירות האלה. נניח ש e_1 קלה יותר מכל קשתות העץ המקורי שיכלו לחבר את שני הרכיבים, אז יכולנו להוסיף את e_1 לעץ המקורי. בכך היה מתקבל מעגל שמכיל לפחות קשת אחת e_2 שמחברת את שתי הקבוצות. היינו מסירים את e_2 ומקבלים עץ קל יותר מהעץ המקורי. זאת סתירה. לכן בעץ המקורי קיימת לפחות קשת אחת שיכולה להחליף את e_1 בעץ החדש מבלי לגרום לעץ להיות כבד יותר.
<u>סיבוכיות</u>: בעץ המקורי היו $|V|-1$ קשתות. כעת הוספנו עוד לכל היותר $|V|$ קשתות. לכן זמן ריצת האלגוריתם היא $O(|V|\log|V|)$.

<u>שאלה 2</u> (מבחינה של פרופ' נוגה אלון ופרופ' עמוס פיאט)
נתונה רשת זרימה $G(V,E)$ עם מקור s ובור t ובה זרימה f שערכה $|f|=1000$. האם בהכרח יש ברשת גם זרימה f' שערכה $|f'|=700$?

כן
נסתכל על רשת שקבוצת צמתיה היא קבוצת הצמתים של הרשת המקורית ובנוסף הצומת t_1 וקבוצת קשתותיה היא קבוצת הקשתות של הרשת המקורית ובנוסף קשת בעלת קיבול 700 מ t ל t_1. ברשת זאת נחפש זרימה מכסימלית מ s ל t_1. מכיון שבכל רשת הערך של זרימה מכסימלית שווה לערך חתך מינימלי, אז ברשת המקורית החתך המינימלי הוא של 1000 . ברשת החדשה החתך המינימלי הוא בגודל 700. לכן הזרימה המכסימלית בו היא בגודל 700. אם נזרים מ s ל t_1 זרימה בגודל 700 אז כל הזרימה עוברת דרך t .

שאלה 3 (**מבחינה של פרופ' יוסי עזר, פרופ' עמוס פיאט ופרופ' מיכה שריר**)
יהא $G(V,E)$ גרף מכוון, ויהיו $s,t \in V$. תארו אלגוריתם יעיל ככל האפשר, שבודק אם קיים מסלול לא פשוט מ- s ל- t. הוכיחו את נכונות האלגוריתם ונתחו את סיבוכיותו.

סיבוכיות: $O(|V|+|E|)$

אלגוריתם:
1. באמצעות שתי הרצות של אלגוריתם DFS נמצא את רכיבי הקשירות החזקה בגרף.
2. באמצעות אלגוריתם BFS (או DFS) נמצא את קבוצת הצמתים אליהם יש מסלול מהצומת s.
3. נסתכל על הגרף שקבוצת קשתותיו היא הקשתות בעלות כיוון הפוך מקשתות הגרף המקורי. בגרף זה נמצא את קבוצת הצמתים אליהם יש מסלול מהצומת t (שוב באמצעות אלגוריתם BFS).
4. התשובה היא חיובית אם ורק אם יש צומת שכלול גם בקבוצה שנמצאה בשלב 2 וגם בקבוצה שנמצאה בצעד 3 ושהוא נמצא ברכיב קשירות חזקה שבו יותר מצומת אחד.

הוכחה:
כדי שיהיה מסלול לא פשוט מ- s ל- t, צריך שיהיה מעגל שאליו יש מסלול מ- s וממנו יש מסלול ל- t. שני צמתים נמצאים על מעגל משותף אם"ם הם באותו רכיב קשירות חזקה. קיום מסלול מהמעגל לצומת t שקול לקיום מסלול מהצומת t למעגל בגרף שבו הפכנו את כיוון הקשתות.

שאלה 4 (**מבחינה של פרופ' אורי צוויק**)
נתונה רשת זרימה מכוונת $G(V,E)$ עם קיבולים $c:E \rightarrow R^+$, ונתונה זרימה מקסימלית f. תאר/י אלגוריתם יעיל ככל האפשר שבודק האם קיימת קשת ברשת שהגדלת קיבולה מאפשרת את הגדלת הזרימה.

סיבוכיות: $O(|V|+|E|)$

אלגוריתם:
נבדוק אם קיים מסלול מצומת המקור לצומת היעד שמשתמש בקשת אחת שהיא רוויה ובמספר כלשהו של קשתות שאינן רוויות. נבצע סריקה דומה לסריקה של BFS, אך צומת יוכל להכנס לתור העתידים לסריקה לא רק פעם אחת אלא לפעמים פעמיים. לגבי כל צומת שאליו נגיע, נסמן אם הגענו אליו בלי שימוש בקשתות רוויות או במסלול שעובר גם בקשת רוויה אחת. לא נכניס לתור צמתים שאליהם הגענו על-ידי שימוש ביותר מקשת רוויה אחת. אם צומת מסומן ככזה שהגענו אליו רק תוך שימוש גם בקשת רוויה וכעת הגענו אליו מבלי שימוש בכזאת, אז נכניס אותו פעם נוספת לתור.

הוכחת נכונות:
אם אין מסלול שמשתמש רק בקשת רוויה אחת, אז צריך להגדיל את הקיבול של לפחות שתי קשתות כדי לקבל מסלול הוספה. אם יש מסלול שמשתמש רק בקשת אחת כזאת אז די להגדיל את הקיבול של קשת זאת. נניח שיש מסלול מתאים באורך סופי, לצמתים שאליהם יש מסלול מתאים באורך 1 בודאי הסריקה תגיע. נניח שלכל צומת שאליו יש מסלול מתאים באורך k מגיעים וגם מגיעים בלי שימוש בקשתות רוויות אם קיים מסלול כזה, אז הצמתים שבמרחק k יכנסו לתור, תוך שמסמנים אם השתמשנו בקשת רוויה. לכן גם לצמתים שאליהם יש מסלול באורך $k+1$ נגיע תוך שנסמנם נכון.

<u>שאלה 5</u> **(מבחינה של פרופ' אורי צוויק)**
יהא $G=(V,E)$ גרף לא מכוון עם פונקצית משקל אי שלילית $w:E\to R^{+}$, מוגדרת על קשתותיו.
תאר/י אלגוריתם יעל ככל האפשר למציאת קבוצת קשתות $E^{'}$ שסכום משקליהן מינימלי כך שבגרף שמתקבל מ- G ע"י זריקת קשתות $E^{'}$ אין מעגלים.

<u>סיבוכיות</u>: $O(|V|+|E|+\min(|E|,|V|)\log(|E|))$

<u>אלגוריתם</u>:
נעבור על קשתות הגרף ונשאיר רק את הצמתים שלפחות קשת אחת נוגעת בהם.
בגרף שמתקבל נריץ בכל רכיב קשירות אלגוריתם למציאת עץ פורש בעל משקל מכסימלי. אוסף הקשתות $E^{'}$ יהיה אוסף הקשתות שלא יכללו בעץ זה. אלגוריתם למציאת עץ פורש מכסימלי יהיה למשל האלגוריתם של פרים למציאת עץ פורש מינימלי שרץ על הגרף שבו לקשת בעלת משקל a בגרף המקורי יינתן משקל $-a$.

<u>הוכחת נכונות</u>:
צריך למעשה להביא למכסימום את סכום משקל הקשתות שאינן מוסרות. מכיון שהמשקולות הן אי שליליות אז כל תוספת של קשת לקבוצת הקשתות הלא מוסרות, מגדילה את משקלה. לכן כל עוד אין בה מעגלים כדאי להוסיף לה קשתות. לכן בכל רכיב קשירות נקבל עץ פורש בעל משקל מכסימלי. עץ פורש בעל משקל W ברכיב קשירות בעל n צמתים בגרף המקורי הוא בעל משקל $-W$ בגרף שבו ביצענו את הטרנספורמציה המוזכרת על משקלי הקשתות. לכן הבעיה שקולה למציאת עץ פורש בעל משקל מינימלי לאחר ביצוע הטרנספורמציה.

<u>הערה</u>
יכולנו גם לחשב עצים פורשים מכסימלים בגרף המקורי.

<u>שאלה 6</u> **(מבחינה של פרופ' אורי צוויק)**
יהא A מערך של n מספרים ממשיים. תאר/י אלגוריתם יעיל ככל האפשר למציאת תת-קבוצה של איברי A שסכומה מקסימלי, תחת האילוץ שאסור לבחור לתת-קבוצה זאת שני איברים סמוכים מ-A.

<u>סיבוכיות</u>: $O(n)$

<u>אלגוריתם</u>:
נשתמש בתכנות דינמי.
יהיו $a[k]$ איברי המערך. נגדיר $F(n+1)=F(n+2)=0$, עבור כל $1\le k\le n$ נחשב את $F(k)$ באופן הרקורסיבי הבא: $F(k)=\max\{F(k+1),a[k]+F(k+2)\}$. לאחר שנחשב את כל הערכים של F , נעבור על פני איברים החל מהראשון כשנגיע לאבר k נבחר אותו אם"ם מתקיים $a[k]+F(k+2)\ge F(k+1)$ ובמקרה זה נעבור ל $k+2$, אם לא נבחר אותו אז נעבור ל $k+1$.

<u>הוכחת נכונות</u>:
בהינתן ערכי הפתרונות האופטימלים עבור כל הקטעים בין p ל n עבור כל $p>k$, אז הפתרון האופטימלי בקטע בין k ל p אומר שיש לבחור את $a[k]$ ולדלג על האיבר שבמקום ה-$k+1$ או לקבל את הפתרון האופטימלי החל ממקום $k+1$.
יש עבור כל k מספר סופי קבוע של השוואות.

<u>שאלה 7</u> **(מבחינה של פרופ' נוגה אלון ופרופ' עמוס פיאט)**
נתון גרף לא מכוון $G=(V,E)$, עם צבע $c(e)$ (לבן או שחור) לכל קשת e , ומשקל שלם $1 \le w(e) \le 100$ לכל קשת e , כשהיצוג ע"י רשימות שכנות ומשקל כל קשת וצבעה מופיעים ליד הופועותיה ברשימות השכנות. תאר/י אלגוריתם יעיל ככל האפשר המוצא מבין כל העצים הפורשים של G המכילים את המספר הגדול ביותר האפשרי של קשתות לבנות, עץ כזה בעל משקל מינימלי.

<u>סיבוכיות</u>: $O(|V|+|E|)$

<u>הסבר</u>:
נפעיל את האלגוריתם של פרים למציאת עץ פורש מינימלי על גרף שבו משקלי הקשתות יהיו פונקציה של משקלי הקשתות בגרף המקורי ושל צבען. קשת לבנה תקבל את משקלה המקורי וקשת שחורה תקבל את משקלה המקורי ועוד $200|V|$ (בחירה שרירותית). כך תהיה עדיפות מוחלטת למספר הקשתות הלבנות ומבין העצים בעלי משקל שווה של קשתות לבנות יועדפו בעלי המשקל הנמוך. את הצמתים שאליהם עדיין לא הגענו נחזיק ב 200 תורים שכל אחד מהם יכלול קבוצה של צמתים בעלי מרחק מינימלי מסוים לקבוצת הצמתים שכבר צברנו במהלך ביצוע האלגוריתם, צומת יוכל לעבור רק לתורים המציינים מרחק נמוך יותר. בכל שלב נצרף לעץ צומת קרוב כמה שיותר. מספר העידכונים הכולל במשך ביצוע האלגוריתם לגבי צמתים שעדיין לא בעץ הוא $O(|e|)$. כל עידכון לוקח לא יותר ממספר קבוע של פעולות.

<u>הערה</u>: סיבוכיות האלגוריתם של קרוסקל לא תהיה כאן לינארית. אמנם בעזרת radix sort אפשר למיין את הקשתות בזמן לינארי, אך הסיבוכיות של union-find אינה לינארית.

<u>שאלה 8</u> **(מבחינה של פרופ' נוגה אלון)**
נתון גרף מכוון $G(V,E)$, ולכל קשת e זוג משקלים $(a(e),b(e))$, שערכו $(0,0)$, $(0,-1)$ או $(-1,0)$.
תארו אלגוריתם יעיל ככל האפשר שבודק האם קיים ב- G מעגל מכוון (לא בהכרח פשוט) בך שסכום רכיבי ה- a שלו שלילי וגם סכום רכיבי ה- b שלו שלילי. הוכיחו את נכונות האלגוריתם וחשבו את סיבוכיותו.

<u>סיבוכיות</u>: $O(|V|+|E|)$

<u>תאור האלגוריתם</u>
על-ידי שתי הרצות של אלגוריתם DFS נמצא את אוסף רכיבי הקשירות החזקה. בשלב הבא נעבור על פני כל קשתות הגרף ולגבי כל קשת נייחס אותה לרכיב קשירות חזקה מסוים אם שני קצוותיה באותו רכיב.
אחר-כך נעבור על-פני רכיבי הקשירות החזקה השונים ונבדוק אם קיים רכיב כזה שבו גם קשת מסוג $(0,-1)$ וגם קשת מסוג $(-1,0)$. אם"ם התשובה לכך היא חיובית אז קיים מעגל כנדרש.

<u>הוכחת נכונות</u>
שתי קשתות שאינן ברכיב קשירות חזקה אחד אינן על מעגל כי יכול להיות מסלול לכל היותר מצמתי אחת משתי הקשתות לצמתי הקשת האחרת. ברכיב קשירות חזקה יש מסלול מכוון מכל צומת לכל צומת, לכן יש מעגל שעובר בכל הצמתים שבקצוות של שתי קשתות.

<u>שאלה 9</u> **(מבחינה של פרופ' יוסי עזר)**

יהא $G(V,E)$ גרף לא מכוון וקשיר, עם משקלים $w:E\rightarrow R^{+}$.

<u>א</u>. תארו אלגוריתם יעיל ככל האפשר אשר בהינתן צומת $u\in V$ מוצא מבין העצים הפורשים המינימלים עץ T שעבורו $d_T(u)$ מקסימלי, כאשר $d_T(u)$ זו דרגת u בעץ T. הוכיחו את נכונות האלגוריתם ונתחו את סיבוכיותו.

<u>סיבוכיות</u>: $O(|E|+|V|\log(|V|))$

<u>אלגוריתם</u>:
באמצעות האלגוריתם של פרים נמצא עץ פורש מינימלי בגרף המושרה על קבוצת הצמתים חוץ מ u.
לקבוצת קשתות העץ שהתקבל נוסיף את כל הקשתות שנוגעות ב u.
בגרף שזו קבוצת קשתותיו נמיין את הקשתות לפי משקלן ונמצא גם את d - ההפרש המינמלי בין משקל זוג קשתות שונות משקל.
נוריד גודל $\frac{d}{2|V|}$ ממשקל כל קשת שנוגעת ב u ולא נשנה את משקלי יתר הקשתות.
הפתרון שנתן יהיה העץ הפורש המינימלי המתקבל על גרף זה.

<u>הוכחה</u>:
לפי פתרון שאלה 1 בקובץ קיים עץ פורש מינימלי שלא משתמש באף קשת שגם לא נוגעת ב u וגם לא כלולה בעץ הראשון שחישבנו. מכיון שניתנה עדיפות לכל קשת על פני כל הקשתות הכבדות ממנה אז העץ המתקבל הוא עץ פורש מינמלי. מכיון שניתנה עדיפות לקשתות הנוגעות ב u אז מבין כל שני עצים שווי משקל ניתנת עדיפות לעץ שבו דרגת u גבוהה יותר.
בשלב השני היו רק סדר גודל של $|V|$ קשתות.

<u>ב</u>. תארו אלגוריתם יעיל ככל האפשר אשר בהינתן שני צמתים $u,v\in V$ מוצא מבין העצים הפורשים המינימלים עץ T שעבורו $d_T(u)-d_T(v)$ מקסימלי. הוכיחו את נכונות האלגוריתם ונתחו את סיבוכיותו.

<u>סיבוכיות</u>: $O(|E|+|V|\log(|V|))$

<u>אלגוריתם</u>:
נשתמש באלגוריתם דומה. בשלב הראשון נמצא עץ פורש מינימלי בגרף המושרה על כל קבוצת הצמתים חוץ מ u ו v. בשלב השני נצרף את u ואת v ואת כך הקשתות הנוגעות בהם. בשלב השני נוריד $\frac{d}{2|V|}$ ממשקל כל קשת שנוגעת ב u ונוסיף $\frac{d}{2|V|}$ למשקל כל קשת שנוגעת ב v (כך לא ישתנה משקל קשת שבין u ל v).

<u>הוכחה</u>:
שוב, כל קשת נשארה קלה יותר מכל הקשתות שהיו קלות ממנה לכן, בשלב השני שוב יתקבל עץ פורש מינימלי. בשלב השני תינתן עדיפות מתאימה בהתאם לנגיעה ב u או ב v.

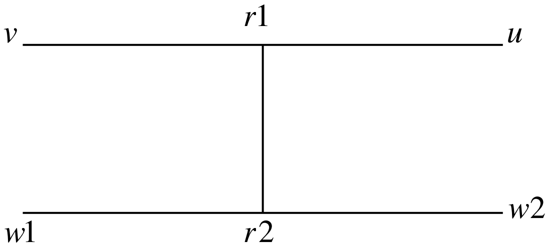
<u>הערה</u>:
ניתן היה גם לפתור באמצעות הרצה אחת של האלגוריתם של פרים תוך מתן עדיפיות לקשתות מבלי חישוב של משקלים חדשים.
הסיבוכיות של אלגוריתם קרוסקל היא יותר גבוהה משל אלגוריתם פרים. לכן, לפני שהרצנו את אלגוריתם קרוסקל, צימצמנו את מספר הקשתות המעומדות להיות חלק מעץ פורש מינימלי.

<u>שאלה 10</u> **(מבחינה של פרופ' עוזי וישקין)**
יהא $G(V,E)$ עץ לא מכוון. <u>המרחק</u> בין שני צמתים ב- G הוא אורך המסלול הקצר ביותר ביניהם, בספירת קשתות. המרחק המכסימלי בין שני צמתים כלשהם בגרף הוא <u>הקוטר</u> של הגרף. תנו אלגוריתם מהיר ככל שתוכלו שמוצא את הקוטר של G. מה סיבוכיות האלגוריתם שנתתם ?
(רמז: יהא v צומת כלשהו בגרף. מה תוכלו לאמור על צומת u שמרחקו מ- v הוא מכסימלי ?)

<u>סיבוכיות</u>: $O(|V|)$

<u>אלגוריתם</u>:
1. נבחר צומת v שרירותי בעץ. מצומת זה נחשב באמצעות אלגוריתם BFS את המרחקים לכל צמתי העץ. נבחר צומת u שעבורו התקבל מרחק כזה מכסימלי.
2. מהצומת u נחשב באמצעות אלגוריתם BFS את המרחקים לכל צמתי העץ. המרחק המכסימלי שיתקבל בשלב הזה הוא קוטר העץ.

<u>נכונות</u>:
צריך למעשה להוכיח שהצומת u הוא קצה של קוטר.
נניח בשלילה ש u אינו קצה של קוטר. אז קיימים זוג צמתים $w1$ ו $w2$ שהמרחק ביניהם גדול מהמרחק של u מכל צומת בעץ.



שני המסלולים מ- v ל- u ומ- $w1$ ל- $w2$ מחוברים על-ידי מסלול מ- $r1$ ל- $r2$. (יתכן ש $r1$ ו $r2$ הם אותו צומת.) המרחק מ- u ל- $r2$ קטן מהמרחק מ- $w1$ ל- $r2$. לכן המרחק מ- u ל- $r1$ קטן מהמרחק מ- $w1$ ל- $r1$. לכן המרחק מ- v ל- u קטן מהמרחק מ- $w1$ ל- v. לכן u לא היה יכול להבחר כבעל מרחק מכסימלי מ- v בשלב 1.
מספר הקשתות בעץ הוא $|V|-1$, לכן הסיבוכיות של BFS היא כאן $O(|V|)$.

<u>שאלה 11</u> (מבחינה של פרופ' נוגה אלון ופרופ' עמוס פיאט)

נתון גרף לא מכוון $G=(V,E)$, מיוצג ע"י רשימות שכנות. תאר/י אלגוריתם יעיל ככל האפשר שיקבע אם קיים תת גרף $H=(V,E')$ של G, שקבוצת צמתיו היא קבוצת הצמתים של G, ו $E'\subseteq E$ מקיימת $|E'|\le 2|V|$, כאשר H עצמו גרף קשיר וחסר גשרים (ז"א: לכל $e\in E'$ גם הגרף H ללא הקשת e הוא קשיר).

<u>סיבוכיות</u>: $O(|V|+|E|)$

<u>הסבר</u>:

באמצעות אלגוריתם DFS נבדוק אם הגרף G הוא גרף קשיר וחסר גשרים. אם הוא לא קשיר או בעל גשרים, אז בודאי אם נסיר ממנו קשתות הוא יהיה בלתי קשיר או בעל גשרים. אם הוא קשיר וחסר גשרים אז יש לו עץ פורש בעל $|V-1|$ קשתות. נראה שניתן להוסיף לעץ לא יותר מ $|V-1|$ קשתות ולקבל גרף קשיר וחסר גשרים. לצומת שהוא עלה בעץ יש קשת שהיא לא קשת בעץ שמחברת אותו לצומת אחר בגרף. נצרף את הקשת הזאת ובכך יווצר תת גרף חסר גשרים שכולל את העלה. כעת בכל שלב שבו עדיין יש צומת שאינו בתת הגרף חסר הגשרים של העלה, נוסיף קשת שמחברת את רכיב זה עם צומת שאינו ברכיב זה ובכך ירד מספר הצמתים שאינם בתת גרף חסר הגשרים של העלה בלפחות 1. בתחילה היה מספר צמתים אלה שווה ל $|V-1|$ לכן דרושות לא יותר מ $|V-1|$ תוספות של קשתות כדי להורידו לאפס.

<u>שאלה 12</u> (מבחינה של פרופ' נוגה אלון)

נתונה רשת זרימה בה הקיבולים של כל הקשתות הם מספרים שלמים וזוגיים, מלבד קשת אחת (u,v) שלה קיבול אי-זוגי. נתונה זרימה מקסימלית f , ונניח שערכה אי-זוגי. האם (u,v) בהכרח רוויה ?

<u>תשובה</u>: כן

<u>הוכחה</u>:

נניח בשלילה שהיא לא רוויה. אם היא לא רוויה אז ניתן להקטין את קיבולה ועדיין לקבל את אותה זרימה מקסימלית. ערך זרימה זאת הוא אי-זוגי. אך זרימה מקסימלית שווה לגודל חתך מינימלי , לאחר השינוי ישארו קיבולי כל יתר הקשתות זוגיים וקיבול קשת זאת לא יהיה מספר שלם אי-זוגי לכן גודל החתך המינימלי לא יהיה מספר אי-זוגי.

<u>שאלה 13</u> (מבחינה של פרופ' מיכה שריר)

נתונה סדרת מספרים ממשיים $X_1,X_2,\cdots,X_n$. תארו אלגוריתם יעיל ככל האפשר המחשב לכל אינדקס j את מספר התת-סדרות העולות ממש המסתיימות ב- x_j (תת-סדרה עולה ממש היא תת-סדרה $X_{a_1},X_{a_2},\cdots,X_{a_i}$, כאשר $a_1<a_2<\cdots<a_i$, המקיימת $X_{a_1}<X_{a_2}<\cdots<X_{a_i}$. היא מסתיימת ב- x_j אם $a_i=j$.)

<u>יעילות</u>: $O(n^2)$

אלגוריתם:
נשתמש בתכנות דינמי. עבור המספר הראשון בסדרה מספר התת-סדרות המתאימות הוא 1. נעבור על כל האינדקסים מונוטונית מהקטן לגדול. עבור כל איבר, מספר התת-סדרות המתאימות שמסתיימות בו יהיה שווה ל 1 ועוד הסכום של מספרי התת-סדרות המתאימות שמסתיימות במספרים בעלי אינדקס נמוך יותר שהם גם קטנים מהמספר עצמו (עוברים על כל המספרים שהם גם בעלי אינדקס נמוך יותר וגם קטנים יותר).

הסבר:
עבור כל j כל תת סדרה מתאימה או שמכילה רק את המספר בעל אינדקס j או שהאיבר הלפני האחרון הוא אחד המספרים הקטנים יותר שהם בעלי אינדקס קטן יותר. זאת חלוקה למקרים זרים לפי זהות האיבר הלפני אחרון.
לגבי כל אינדקס j עוברים על פני לא יותר מ n אינדקסים קטנים יותר ומסכמים לא יותר מ n מספרים.

שאלה 14 **(מבחינה של פרופ' נוגה אלון)**
נתון גרף לא מכוון $G(V,E)$ מיוצג ע"י רשימות שכנות, נתון משקל שלם $1 \le w(e) \le |V|$ לכל קשת $e \in E$, ונתונים זוג צמתים $s,t \in V$. תאר/י אלגוריתם יעיל ככל האפשר המוצא מסילה מ- s ל- t כך שהמשקל המירבי של קשת במסילה הוא קטן ככל האפשר.

סיבוכיות: $O(|V|+|E|)$

הסבר:
האלגוריתם יורכב משני שלבים עיקריים בשלב הראשון נמצא אוסף קשתות בעלות משקל מכסימלי קטן ככל היותר שמאפשרות קשירות בין s ל t. בשלב השני נבצע BFS בגרף שאוסף קשתותיו נמצא בשלב הראשון. ה- BFS יתחיל בצומת s עד הגעה לצומת t כאשר לא נסתכל על המשקל של כל קשת באוסף זאת כי כבר ברור שבשביל לגרום ל- s ול- t להיות ברכיב קשירות אחד, צריך קשתות במשקל של לפחות המשקל המכסימלי של קשתות האוסף.
<u>השלב הראשון</u>: נעבור על-פני קשתות הגרף ונכניס את כל אחת מהן לרשימה של קשתות בעלות משקל שווה לשלה (יהיו $|V|$ רשימות של קשתות). נסמן את הצומת s שהגענו אליו. נסמן את יתר הצמתים ככאלה שבשלב זה לא הגענו אליהם. לכל צומת נחזיק תור של קשתות שנוגעות בו ושעליהן צריך לעבור כאשר נגיע לצומת זה. בתחילה יהיו בתורים רק הקשתות בעלות משקל 1. נחזיק תור של צמתים שאליהם הגענו ושצריכים להיסרק. כל עוד לא הגענו לצומת t נסרוק צמתים שבתור זה. צומת שהתור שלו יתרוקן יצא לפחות באופן זמני מהתור של הצמתים שצריכים להיסרק. כאשר התור של הצמתים יתרוקן מבלי שהגענו לצומת t אז נוסיף קשתות חדשות לתורי הקשתות של צמתים שבקצוותיהם ומבין הצמתים האלה נוסיף את אלה שאליהם כבר הגענו לתור הצמתים שצריכים להיסרק. הקשתות החדשות שנוסיף הן הקשתות מהרשימה של הקשתות הכי קלות מבין אלה שעדיין לא עברנו עליה. צמתים שאליהם נגיע לראשונה במהלך הסריקה גם יכנסו לתור הצמתים שאותם צריך לסרוק (לגבי כל צומת כזה, יתכן שכבר יש בתור הקשתות שלו קשתות שעליהן צריך לעבור, הקשתות האלה נצברו כבר לפני שהגענו לצומת זה).

הערה: ניתן לפתור גם בדרכים שונות לחלוטין.

<u>שאלה 15</u> **(מבחינה של פרופ' נוגה אלון ופרופ' מיכה שריר)**

נתון גרף מכוון $G=(V,E)$, המיוצג ע"י רשימות שכנות.

תארו אלגוריתם יעיל ככל האפשר, המחשב את הקבוצה $U \subseteq V$ של הצמתים u, בעלי התכונה שלכל $v \in V$ קיימת מסילה מכוונת מ- u ל- v.

<u>יעילות</u>: $O(|E|+|V|)$

<u>אלגוריתם</u>:

באמצעות שתי הרצות של אלגוריתם DFS נמצא את רכיבי הקשירות החזקה בגרף.

נחשב את גרף העל של הגרף G.

לגבי גרף העל, נסתכל על הגרף שאוסף קשתותיו הן בעלות כיוון הפוך לקשתות הגרף המקורי, בגרף זה נבחר צומת שרירותי וממנו נעשה BFS לצורך מציאת עלה- w.

בגרף העל המקורי נבצע BFS מהצומת w. אם"ם בסריקה זאת נגיע לכל צמתי הגרף, אז צמתי רכיב הקשירות שמייצג w הם הקבוצה U. אחרת הקבוצה ריקה.

<u>הסבר</u>:

מכל צומת בקבוצה U צריך להיות מסלול לכל צומת אחר בקבוצה U. לכן קבוצת הצמתים U היא רכיב קשירות חזקה. בגרף שכיוון קשתותיו הפוך לכיוון המקורי דרוש שיהיה מסלול מכל צומת לצמתי U ושמצמתי U לא יהיה מסלול לצמתים אחרים כי אחרת הם היו חלק מאותו רכיב קשירות חזקה. לכן BFS בגרף שכיוון קשתותיו הפוך לכיוון המקורי של קשתות גרף העל חייב להגיע לצומת שמייצג את הקבוצה U ושזה יתגלה כעלה.

<u>שאלה 16</u> **(מבחינה של פרופ' נוגה אלון ופרופ' מיכה שריר)**

נתונה מחרוזת T של n תווים ותבנית P של m תווים. תארו אלגוריתם יעיל ככל האפשר הבודק אם קיימות מחרוזות x, y ורישא (לא ריקה) P' של P, כך ש- $T = xP'y$ וגם מתקיים $|y| < 2|P'|$

<u>יעילות</u>: $O(\min(m,n))$

<u>אלגוריתם</u>:

נתייחס לתבנית שהיא התבנית המקורית אם $n \geq m$ ואחרת היא רק n התווים הראשונים בתבנית T.

נריץ את אלגוריתם KMP למציאת כל ההופעות של התבנית במחרוזת חלקית של T שהיא $\min(n,3m)$ התווים האחרונים של T : עבור כל מקום k במחרוזת T המקורית, נקבל את $\max_k$- אורך הרישא הארוכה ביותר שמסתיימת בו.

הדרישה מתקיימת אם"ם קיים k כך ש $2\max_k > n-k$.

<u>הסבר</u>:

אם יש התאמה של רישא מסוימת במקום מסוים, אז תמיד אפשר לבחור את x ואת y שיהיו זהים לתתי המחרוזות שלפני הרישא המתאימה ואחרי הרישא המתאימה. קיום k כך ש $2\max_k > n-k$ הוא בדיוק הדרישה הנחוצה. הדרישה שיתקיים $|y| < 2|P'|$ אומרת שההתאמה החלקית או המלאה חייבת להיות ב $3m$ המקומות האחרונים של T (כי התאמה היא באורך של לא יותר מ m).

לגבי היעילות: מריצים KMP כאשרי אורך המכסימלי מבין המחרוזות והתבנית הוא $\min(n,3m)$.

<u>שאלה 17</u> **(מבחינה של פרופ' נוגה אלון ופרופ' מיכה שריר)**

נתון גרף לא מכוון $G=(V,E)$ המיוצג ע"י רשימות שכנות, וצומת $s\in V$. לכל קשת $e\in E$ יש משקל שלם $w(e)$ (חיובי או שלילי). תארו אלגוריתם יעיל ככל האפשר המחשב, לכל צומת $v\in V$, את המשקל המינימלי של מסלול מ- s ל- v בעל <u>מספר זוגי</u> של קשתות. (הניחו שב- G אין מעגלים שליליים.)

<u>יעילות</u>: $O(|E|+|V|\log|V|)$ כמו דיקסטרה או אם לא מניחים שהקשתות אי-שליליות אז $O(|V||E|)$ כמו בלמן פורד

<u>הערה</u>: במקרה הזה היתה הנחיה מיוחדת שקשת שעוברים עליה פעם בכל כיוון היא מעגל באורך 2. לכן אין למעשה קשתות בעלות משקל שלילי.

<u>אלגוריתם</u>:

נסתכל על גרף דו-צדדי שבו לכל צומת מהגרף המקורי יש עותק בכל צד. עבור כל זוג צמתים i,j משני צדדים שונים, יש בינהם קשת אם בגרף המקורי יש קשת בין שני הצמתים שהם מייצגים. משקל קשת זאת זהה למשקל הקשת מהגרף המקורי. בגרף זה נמצא את המרחקים מעותק אחד של הצומת s לכל יתר הצמתים שבצד זה. אלה המרחקים הדרושים.

<u>הסבר</u>:

מכיון שהגרף שנבנה הוא דו-צדדי אז כל המסלולים בעלי אורך זוגי שמתחילים בצד אחד מסתיימים באותו צד. לכל מסלול בעל אורך זוגי בגרף המקורי, יש מסלול מתאים בגרף זה. כל מסלול בגרף זה מייצג מסלול בגרף המקורי.

מכיון שבגרף המקורי אין מעגלים בעלי אורך שלילי, אז גם בגרף הדו-צדדי אין מעגלים בעלי אורך שלילי. מספר צמתי הגרף הוא כפליים מספר צמתי הגרף המקורי ומספר קשתות הגרף הוא כפליים מספר קשתות הגרף המקורי. מכאן מתקבלת היעילות הרשומה.

שאלה 18 **(מבחינה של פרופ' נוגה אלון ופרופ' מיכה שריר ומבחינה של פרופ' אורי צוויק)**
נתון גרף מכוון $G(V,E)$, המיוצג ע"י רשימות שכנות.
תארו אלגוריתם יעיל ככל האפשר, המוצא קבוצת צמתים לא ריקה $U \subseteq V$ מגודל מינימלי אפשרי, כך שאין שום קשת מכוונת היוצאת מצומת של U לצומת שאינו ב- U.

יעילות: $O(|V|+|E|)$

אלגוריתם:
1. באמצעות שתי הרצות של אלגוריתם DFS נמצא את רכיבי הקשירות החזקה בגרף.
2. נמצא את כל רכיבי הקשירות החזקה שמהם לא יוצאות קשתות לצמתים שמחוץ להם.
3. לקבוצה U נבחר את כל צמתי הרכיב בעל מספר מינימלי של צמתים מבין הרכיבים שנמצאו בשלב 2.

הסבר:
מכל צומת יש מסלול לכל צומת שבאותו רכיב קשירות חזקה. לכן אם בוחרים צומת מסוים, אז כל הצמתים מרכיב הקשירות החזקה שלו צריכים להבחר. מכל רכיב קשירות חזקה יש מסלול לרכיב קשירות חזקה שמימנו אין קשתות אל מחוצה לו. לכן חייבים לבחור לפחות רכיב קשירות חזקה אחד שממנו אין יציאות. בחרנו רכיב כזה שהוא בעל גודל מינימלי.

שאלה 19 **(מבחינה של פרופ' נוגה אלון ופרופ' מיכה שריר ומבחינה של פרופ' אורי צוויק)**
יהא $G=(V,E)$ גרף לא מכוון מיוצג ע"י רשימות שכנות, עם משקל שלם $1 \le w(e) \le |E|$ לכל קשת $e \in E$ כך שאין אף זוג קשתות בעלות אותו משקל.
קשת נקראת <u>כבדה</u> אם היא בעלת משקל מקסימלי במעגל פשוט כלשהו ב- G.
תארו אלגוריתם יעיל ככל האפשר שימצא את כל הקשתות הכבדות בגרף.

יעילות: $O(|V|+|E|\cdot\alpha(|E|,|V|))$

אלגוריתם:
באמצעות הרצה בודדת של האלגוריתם של קרוסקל על כל הגרף נמצא עצים פורשים מינימלים בכל רכיבי הקשירות של הגרף.
הקשתות הכבדות הן הקשתות שלא יבחרו ליער זה.

הסבר:
כאשר כל המשקולות שונים, הקשתות שלא נבחרות כאשר מריצים את אלגוריתם קרוסקל הן הקשתות שסוגרות מעגל עם קשתות בעלות משקל קטן ממשקלן.
במקרה המתואר ניתן למיין את הקשתות לפי אורכן בזמן $O(|E|+|V|)$.
לכן כאן השלב שצורך יותר זמן הוא השלב השני באלגוריתם של קרוסקל.

<u>שאלה 20</u> **(מבחינה של פרופ' נוגה אלון ופרופ' מיכה שריר ומבחינה של פרופ' אורי צוויק)**
נתון גרף מכוון $G(V,E)$, המיוצג ע"י רשימות שכנות עם משקל שלם $w(e)$ (חיובי או שלילי) לכל קשת $e \in E$. נתון שאין ב- G מעגל שלילי.
תארו אלגוריתם יעיל ככל האפשר המוצא לכל $v \in V$ את המשקל הקטן ביותר של מסילה מכוונת (בעלת מספר קשתות כלשהו) המתחילה ב- v.

<u>יעילות</u>: $O(|E||V|)$

<u>אלגוריתם</u>:
נמצא את הגרף בעל אותה קבוצה של צמתים ובעל קשתות בכיוון הפוך מהקשתות המקוריות.
לגרף זה נוסיף את צומת s וממנו נוסיף קשתות בעלות משקל אפס לכל אחד מצמתי הגרף.
בגרף שהתקבל נמצא באמצעות אלגוריתם בלמן פורד את משקלי המסילות הקצרות ביותר מהצומת s לכל אחד מצמתי הגרף.
לגבי כל צומת, המרחק שהתקבל מהצומת s בגרף שבנינו, הוא המרחק המינימלי מצומת זה בגרף המקורי.
אם רוצים להימנע ממסילות ריקות ללא קשתות אז משכפלים שני עותקים של צמתי הגרף, הקשתות בכיוון הפוך יהיו בין הנציגים של הצמתים שבאותו עותק וגם בין צמתי העותק הראשון לצמתי העותק השני (רק בכיוון זה). מהצומת s יצאו הקשתות שאותן הוספנו רק לנציגי העותק הראשון. המרחק לצומת יוגדר כמרחק לנציג שלו בעותק השני.

<u>הסבר</u>:
המרחק מצומת s לצומת v בגרף שבנינו שווה למרחק הכי קטן מצומת אחר לצומת v.
זה שווה למרחק הכי קטן מצומת v לאיזשהו צומת בגרף המקורי. על-ידי היפוך כיוון הקשתות והוספות רק קשתות יוצאות מהצומת s לא יצרנו מעגלים בעלי אורך שלילי (לא ניצור כלל מעגלים חדשים). גם אם נשכפל את צמתי הגרף לשני עותקים לא ניצור מעגלים חדשים. המסילות לצמתי העותק השני עוברות לפחות קשת אחת בגרף המקורי.

<u>שאלה 21</u> **(מבחינה של פרופ' נוגה אלון וד''ר איריס רוזנבלום ומבחינה של פרופ' מיכה שריר)**
נתון גרף מכוון $G=(V,E)$ המיוצג ע"י רשימות שכנות ונתונה קבוצת צמתים $U \subseteq V$. תארו אלגוריתם יעיל ככל האפשר שיקבע אם יש מסילה מכוונת (לאו דווקא פשוטה; קרי ייתכן שתעבור בצמתים וקשתות יותר מפעם אחת) שמבקרת בכל הצמתים ב- U.

<u>יעילות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:
נחשב את רכיבי הקשירות החזקה. נמצא מיון טופולוגי של גרף העל.
בגרף העל נבצע DFS החל מהרכיב שממנו יש מסלולים לכל יתר הרכיבים.
יש מסלול כנדרש אם"ם כל רכיבי הקשירות החזקה שבהם יש ייצוג לצמתים מהקבוצה U הם על מסלול פשוט בסריקה שמצאנו. אם אין מסלול כזה אז זה אומר שבין לפחות שני צמתים מ U אין מסלול באף לא כיוון כי ברור שבכיוון ההפוך לסריקה אין מסלול כזה. בכל רכיב קשירות חזקה יש מסלול מכל צומת לכל צומת ובחזרה ולכן ניתן גם לצאת מכל צומת שלו אל רכיב קשירות חזקה אחר. בסריקה אפשר לקבל סדרים שונים. כדי להתגבר על זה נבדוק אם מכל צומת שמייצג צומת שבו צמתים מיוחדים יש מסלול לצומת שהוא הבא אחריו במיון הטופולוגי שבו צמתים מיחדים. אם עבור כל שני זוגות עוקבים זה מתקיים אז יש מסלול כנדרש. נדאג שנגיע לצומת הבא לפני שנסוגונו סופית מהקודם.

<u>שאלה 22</u> **(מבחינה של פרופ' נוגה אלון ופרופ' מיכה שריר ומבחינה של פרופ' אורי צוויק)**

נתון גרף לא מכוון דו-צדדי $G=(V,E)$ המיוצג ע"י רשימות שכנות.

תארו אלגוריתם יעיל ככל האפשר המוצא ב- G תת גרף בעל מספר מירבי של קשתות, שבו לכל קודקוד דרגה לכל היותר 3.

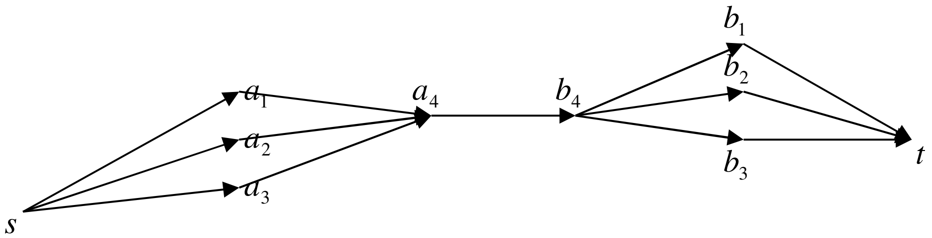
<u>יעילות</u>: $O\left(|E||V|^{2/3}\right)$

<u>אלגוריתם</u>:

נבנה רשת זרימה שעליה נפעיל אלגוריתם זרימה.

לכל צומת מהגרף המקורי יהיו ארבעה צמתים נציגים. נוסיף שני צמתים s ו t לגרף. יהיו קשתות מכוונות מ s לשלושה עותקים של כל צומת בצד הראשון. תהיה קשת מכל אחד מנציגים אלה אל הנציג הרביעי של אותו צומת. יהיו קשתות מכוונות משלושה נציגים של כל צומת מהצד השני אל הצומת t. מכל נציג רביעי יהיו קשתות מכוונות לשלושת הנציגים האחרים של אותו צומת. עבור כל זוג צמתים שיש ביניהם קשת בגרף המקורי תהיה קשת מכוונת מהנציג הרביעי של הצומת שבצד הראשון אל הנציג הרביעי של הצומת שבצד השני. לכל הקשתות יהיה קיבול של יחידה אחת. נפעיל את אלגוריתם דיניץ אלגוריתם למציאת זרימה מכסימלית מ s ל t. זוגות הצמתים מהגרף המקורי שביניהם תהיה קשת הם הצמתים שבין הנציגים הרביעיים שלהם יש זרימה בגרף ברשת שבנינו.

<u>הערה</u>: נייצג ברשת רק צמתים שלהם דרגה לפחות 1 בגרף המקורי.



בשרטוט נתונה רשת שמייצגת בניה לפי גרף מקורי שבו שני צמתים a ו b שביניהם יש קשת.

<u>הסבר</u>:

בכל צומת יכולה לעבור זרימה של לכל היותר שלוש יחידות. הקישור דרך הנציגים הרביעיים דואג לכך שלא תהיה זרימה של יותר מיחידה אחת בין שני צמתים. זאת היא רשת זרימה שבה מספר הצמתים הוא באותו סדר גודל של מספר הצמתים שבגרף המקורי ומספר הקשתות הוא בסדר גודל של מספר הקשתות שבגרף המקורי (לא יצגנו צמתים שלהם דרגה אפס בגרף המקורי). זאת היא רשת זרימה עם קיבולים של יחידה אחת וללא קשתות מקבילות.

<u>שאלה 23</u> **(מבחינה של פרופ' נוגה אלון ופרופ' מיכה שריר)**
נתונה מחרוזת T של n תווים ותבנית P של m תווים, $|T|=n$, $|P|=m<n$.
תארו אלגוריתם יעיל ככל האפשר הבודק האם קיים k שלם כך ש-
$$T = P_{i_1} P_{i_2} \dots P_{i_k}$$
כאשר כל P_{i_j} הוא רישא של P, ואם קיים k כזה הוא מוצא את <u>ה-k הקטן ביותר</u> עבורו זה מתקיים.

<u>יעילות</u>: $O(n)$

<u>אלגוריתם</u>:
<u>השלב הראשון</u>: נריץ את אלגוריתם KMP . עבור כל מקום במחרוזת נקבל את הרישא הארוכה ביותר המסתיימת בו.
<u>השלב השני</u>: אם במקום האחרון במחרוזת מסתיימת רישא, אז נבחר את הרישא הארוכה ביותר המסתיימת במקום זה. נקצץ מהמחרוזת את הסיפא שחופפת לרישא הארוכה ביותר שנמצאה. נמשיך בתהליך דומה בחלק של המחרוזת שנותר. נספור את מספר הפעמים שנקצץ סיפא. הכיסוי על-ידי רישות אפשרי אם באיזשהו שלב נגיע למצב שקיצצנו כבר את כל המחרוזת.

<u>הסבר</u>:
מבלי לפגוע בקיומו של פתרון וגם כדי להביא למינימום את מספר הרישות שבכיסוי, ניתן לבחור בכל שלב ברישא הארוכה ביותר האפשרית של התבנית שתכסה סיפא של המחרוזת. אם קיים פתרון אחר שמשתמש ברישא קצרה יותר, אז ניתן לקצץ בפתרון זה חלק מרישות קודמות, כך שנאריך את הרישא האחרונה על חשבון הקודמות ובסך הכל נכסה את אותו חלק של המחרוזת.
סיבוכיות השלב הראשון היא הסיבוכיות של אלגוריתם KMP . כמובן לא נקצץ חלקים מהמחרוזת יותר מ n פעמים.

<u>שאלה 24</u> **(מבחינה של פרופ' מיכה שריר)**
תארו אלגוריתם יעיל ככל האפשר שבודק, בהינתן שתי מחרוזות S,T באורך n כל אחת, האם T היא הזזה ציקלית של S (לדוגמא car ו arc הן הזזות ציקליות אחת של השניה).

<u>יעילות</u> $O(n)$

<u>אלגוריתם</u>
T_2 יהיה טקסט שיורכב מרצף של שתי מחרוזות T אחת אחר השניה. נריץ אלגוריתם לבדיקה אם S מופיע ב T_2. יש הזזה ציקלית אם S יופיע ב T_2. אם מתייחסים רק להזזות ממש ולא כוללים בהן זהות בין מחרוזות, אז בודקים רק אם S מופיע בטקסט של $2n-2$ אותיות שלא כולל את האות הראשונה והאחרונה של T_2 .

<u>הסבר</u>
אם S היא הזזה ציקלית של T ב k מקומות, אז S יופיע ב T_2 החל ממקום $k+1$ (אם S ו T הן זהות אז הוא יופיע החל ממקום 1).

<u>שאלה 25</u> **(מבחינה של פרופ' נוגה אלון)**

יהא $G=(V,E)$ גרף קשיר, לא מכוון, עם מחיר לכל קשת.

הוכח או הפרך: אם $G_1=(V,E_1)$ ו $G_2=(V,E_2)$ הם שני גרפים קשירים על אותה קבוצת צמתים V עם מחיר לכל קשת, ו $T_1=(V,F_1)$ עץ פורש מינימלי ל- G_1, $T_2=(V,F_2)$ עץ פורש מינימלי ל- G_2 אזי עץ פורש מינימלי בגרף $(V,F_1\cup F_2)$ הוא גם עץ פורש מינימלי של $G=(V,E_1\cup E_2)$.

(נניח כאן כי $E_1\cap E_2=\phi$).

<u>פתרון</u>

נראה שהטענה נכונה. נניח שיש עץ פורש מינימלי של G שבו יש קשת e שהיא ב $E_1\cup E_2$ אך לא ב $F_1\cup F_2$. נניח בלי הגבלת הכלליות שהיא ב E_1. נראה שיש עץ פורש מינימלי אחר של G שבו היא לא נמצאת ובמקומה נמצאת קשת ששיכת ל F_1 כאשר יתר קשתות העץ לא הוחלפו. נסיר אותה מהעץ הנתון של G. בכך יווצרו שני רכיבי קשירות בגרף. הקשת e לא נכללת בעץ הפורש T_1 לכן בהכרח היא סוגרת מעגל עם קשתות מ F_1 שאף אחת מקשתות מעגל זה היא לא יותר כבדה מ e. מבין קשתות המעגל יש לפחות עוד קשת אחת שמחברת את שני רכיבי הקשירות שנוצרו עם הסרת e מהעץ של G. נוסיף את אחת מהקשתות האלה לעץ במקום e ובכך נקבל עץ שסכום משקל קשתותיו לא יותר גדול מהמקורי ולכן גם הוא עץ פורש מינימלי. כך נוכל להחליף אחת אחר השניה את כל קשתות העץ של G שאינן ב $F_1\cup F_2$ בקשתות שהן כן ב $F_1\cup F_2$.

<u>שאלה 26</u> **(מבחינה של פרופ' לאה אפשטיין וד"ר מירי פרייזלר)**

נתון גרף מכוון $G=(V,E)$, כאשר $V=\{1,2,\cdots,n\}$, קודקוד s ב- V, ונתונה פונקצית משקל אי שלילית $w:E\to R^+\cup\{0\}$.

המסלול $\{x_0,x_1,\cdots,x_r\}$ יקרא **מונוטוני** אם $x_0<x_1<\cdots<x_r$ או $x_0>x_1>\cdots>x_r$.

תארו אלגוריתם **יעיל** למציאת משקל המסלולים **המונוטונים** הקלים ביותר מ- s לכל קודקוד.

הסבירו את נכונות האלגוריתם וחשבו את סבוכיותו.

<u>סבוכיות</u> $O(|E|+|V|)$

<u>תאור האלגוריתם</u>

<u>שלב ראשון</u>: נמחק מהגרף את כל הקשתות שמחברות צמתים בעלי אינדקס קטן מהאינדקס של s עם צמתים בעלי אינדקס גדול משל s או להפך.

<u>שלב שני</u>: נשתמש בתכנות דינאמי. המרחק לצומת s הוא אפס. עבור הצמתים בעלי אינדקס גדול מהאינדקס של s: נעבור על הצמתים האלה בסדר עולה. עבור כל צומת v כזה המרחק המבוקש יהיה המינימום על-פני המרחק לצומת שכן w בעל אינדקס קטן יותר (אך לא קטן משל s) + אורך הקשת מהצומת w ל v. עבור הצמתים בעלי אינדקס קטן מהאינדקס של s: נעבור על הצמתים האלה בסדר יורד. עבור כל צומת v כזה המרחק המבוקש יהיה המינימום על-פני המרחק לצומת שכן w בעל אינדקס גדול יותר (אך לא גדול משל s) + אורך הקשת מהצומת w ל v.

<u>הסבר נכונות</u>

אין מסלול מונוטוני שמתחיל בצומת s ועובר גם בצמתים בעלי אינדקס גבוה יותר וגם בצמתים בעל אינדקס נמוך יותר. כל מסלול מ s לצומת שונה מ s עובר רק דרך צמתים בעלי אינדקס קרוב יותר לזה של s. לגבי כל צומת, בהנחה שלגבי כל הצמתים בעלי אינדקס קרוב יותר ל s חשבנו את המרחקים האופטימלים אז במעבר על כל הקשתות שנכנסות אליו נמצה את כל המסלולים הנכנסים אליו.
על כל צומת ועל כל קשת עוברים רק מספר סופי של פעמים.

<u>שאלה 27</u> **(מבחינה של פרופ' לאה אפשטיין וד''ר מירי פרייזלר)**

נתון גרף לא מכוון, קשיר ופשוט $G=(V,E)$, פונקצית משקל $w:E\rightarrow R^{+}\cup\{0\}$, קודקוד s ב- V, ועץ פורש T של הגרף G.
תארו אלגוריתם **יעיל**, הבודק אם T הוא עץ מסלולים קלים ביותר מ- s לכל קודקוד בגרף. הסבירו את נכונות האלגוריתם וחשבו את סבוכיותו.

<u>סבוכיות</u> $O(|E|+|V|)$

<u>תאור האלגוריתם</u>

<u>שלב ראשון</u>: נחשב את המרחקים על העץ מהצומת s אל כל צמתי הגרף. נסרוק את העץ ב BFS החל מ s ולגבי כל צומת w, המרחק מ- s יוגדר כמרחק לצומת שממנו הגענו לצומת w בתוספת אורך הקשת ביניהם.
<u>שלב שני</u>: נעבור על פני כל קשתות הגרף. עבור כל קשת נבדוק אם קיים צומת w מבין שני קצוותיו שהמרחק על העץ לקצה השני של הקשת מ- s בתוספת אורך הקשת שביניהם קטן מהמרחק על העץ לצומת w. אם לפחות פעם אחת ימצא צומת כזה אז העץ איננו עץ מרחקים מינימלים. אחרת הוא כן.

<u>הסבר נכונות</u>

בשלב הראשון נמצא את כל המרחקים הקצרים ביותר על העץ כי לכל צומת יש רק מסלול אחד שמוביל אליו מ- s על העץ.
לגבי השלב השני: אם קיים צומת שלגביו העץ לא נותן מרחק אופטימלי אז יש מסלול קצר יותר לצומת. במסלול זה יש צומת ראשון אליו העץ לא נותן מרחק מינימלי.כאשר נטפל בקשת שבין הצומת הזה להורה שלו במסלול נקבל אי שיוויון שיראה שהעץ לא אופטימלי.
סבוכיות החלק הראשון היא $O(|V|)$ כי בעץ יש רק $V-1$ קשתות. סבוכות החלק השני היא $O(|V|+|E|)$ כי עוברים על פני כל קשת מספר סופי של פעמים.

<u>שאלה 28</u> **(מבחינה של פרופ' לאה אפשטיין וד''ר מירי פרייזלר)**
נתונה רשת זרימה, $G=(V,E)$, מקור s, בור t, ופונקציית קיבול $c: E \to R^+ \cup \{0\}$; כמו-כן נתונה קשת $e=(u,v)$ ב- E. תארו אלגוריתם **יעיל**, המוצא מבין כל הזרימות המקסימליות מ- s ל- t, זרימה מקסימלית f ברשת, עבורה $f(e)$ היא הזרימה המקסימלית האפשרית על הקשת e.
הסבירו את נכונות האלגוריתם וחשבו את סבוכיותו.

<u>סבוכיות</u> של אלגוריתם לחשוב זרימה מקסימלית למשל דיניץ: $O(|V|^2|E|)$

<u>תאור האלגוריתם</u>
<u>שלב ראשון</u>: נחשב זרימה מקסימלית מ- s ל- t.
<u>שלב שני</u>: לאחר שהתקבלה הזרימה המקסימלית בשלב הראשון, נחשב רשת חדשה. נוסיף לצמתים המקוריים שני צמתים u_1 ו v_1. יהיה אוסף חדש של קשתות. תהיה קשת מ- v_1 ל- v ותהיה קשת מ- u ל- u_1. הקיבול בכל אחת מקשתות אלה יהיה שווה למקסימום הזרימה שאפשר להוסיף בקשת (u,v) ביחס לזרימה של השלב הראשון, כך למשל אם החסם העליון הוא 8 יחידות ובשלב הראשון הזרמנו 5 יחידות אז לקשתות החדשות יהיה קיבול 3. את הקשת המקורית בין u ל v ננתק. עבור כל זוג צמתים אחרים שמחוברים בקשת ברשת המקורית, נקבע קשתות בכל אחד משני הכיוונים שקיבולן יהיה שווה למקסימום הזרימה שאפשר להוסיף בכיוון זה ביחס לזרימה של סעיף א', כך למשל אם בקשת ניתן להזרים עד 9 יחידות וכעת מוזרמים בה 7 יחידות, אז בכיוון אחד תהיה קשת בעלת קיבול 7 וקשת שניה בעל קיבול 2.
כעת נחשב זרימה מקסימלית מ- v_1 ל- u_1.
הזרימה המבוקשת הסופית על כל קשת תהיה שווה לסכום הזרימה עליה בשלב הראשון והזרימה בין שני קצוותיה בשלב השני. הזרימה על הקשת e תהיה שווה לסכום הזרימה עליה בשלב הראשון והזרימה מ- v_1 ל- v בשלב השני.

<u>הסבר נכונות</u>
בשלב השני אנו משמרים את הזרימה על כל הצמתים מבלי לחרוג מהקיבולים המותרים, כך שנשמרת הזרימה המקסימלית מ- s ל- t. אך במסגרת זו מוגדלת ככל האפשר הזרימה על הקשת e. אי אפשר להשיג זרימה יותר גדולה מ- s ל- t מאשר זאת שהושגה בשלב הראשון. על הצמתים שהם לא u ולא v חייבים לשמר את המאזן בשלב השני, כך שמה שנכנס אליהם גם יוצא מהם. כמוכן, כדי להזרים יותר מאשר בשלב הראשון מ- u ל- v, צריך שיכנס ל- u יותר וצריך שיצא מ- v. מה שנכניס יותר ל- u זה מה שיזרום בקשתות החדשות בשלב השני.

<u>שאלה 29</u> **(מבחינה של פרופ' מיכה שריר)**

יהא $G=(V,E)$ גרף מכוון עם משקלות חיוביים על הקשתות, המיצג ע"י רשימות שכנות. הגרף מייצג מפת כבישים, כשמשקל כל קשת הוא אורך הקשת שהיא מייצגת.
בחלק מהצמתים נמצאות תחנות דלק.
תארו אלגוריתם יעיל ככל האפשר המוצא לכל צומת בגרף תחנת דלק הכי קרובה ואת המרחק אליה.

<u>יעילות</u> $O(|E|+|V|\log|V|)$

<u>אלגוריתם</u>

נוסיף צומת s שיהיו לו רק קשתות שיוצאות לצמתים שמייצגים תחנות דלק. נהפוך את הכיוון של יתר הקשתות. נריץ את האלגוריתם של דיקסטרה למציאת מסלולים לכל הצמתים מצומת s. עבור כל צומת, המשקל שימצא אליו הוא המרחק ממנו לתחנת הדלק הקרובה ביותר אליו. בעץ שהתקבל, נמצא עבור כל צומת שמייצג תחנת דלק, את קבוצת הצמתים שניתן להגיע אליה ממנו. לגבי כל צומת, תחנת הדלק הכי קרובה היא התחנה שמייצג הצומת שממנו הגענו אליו.

<u>הסבר</u>

מסלול מצומת אל תחנת דלק הוא מסלול מתחנת דלק לאותו צומת על הגרף שכיוון קשתותיו הפוך. לגבי כל צומת, המסלול לצומת s עובר בתחנת הדלק הקרובה ביותר אליו. האלגוריתם של דיקסטרה יוצר עץ מסלולים ולכן לגבי כל צומת, יש מסלול רק מתחנת דלק אחת.

<u>שאלה 30</u> **(מבחינה של פרופ' מיכה שריר)**

נתון גרף מכוון וקשיר $G=(V,E)$ עם משקלות חיוביים על הקשתות, המיוצג ע"י רשימות שכנות.
לכל $t>0$ נגדיר את $E(t)$ להיות קבוצת הקשתות שמשקלן הוא לכל היותר t. תארו אלגוריתם יעיל ככל האפשר המחשב את t המינימלי עבורו $G(t)=(V,E(t))$ הוא קשיר.

<u>יעילות</u> $O(|E|+|V|\log|V|)$

<u>אלגוריתם</u>

נבחר צומת שרירותי. נסמן אותו ככזה שהגענו אליו. נסמן את יתר הצמתים ככאלה שלא הגענו אליהם עדיין. בדומה למה שמבוצע באלגוריתם של פרים, נצרף בכל שלב לקבוצת הצמתים שהגענו אליה צומת אחד. זה יהיה הצומת שלו קשת בעלת משקל מינימלי לצומת שכבר הגענו אליו. נמשיך בתהליך עד שנגיע לכל הצמתים. כעת נהפוך את כיוונן של כל קשתות הגרף ושוב נתחיל באותו צומת שרירותי ובתהליך דומה נוסיף צמתים עד שנגיע לכל צמתי הגרף. t יהיה שווה למכסימום בין משקליהן של הקשתות הכבדות ביותר שנמצאו בשני התהליכים.

<u>הסבר</u>

אם יש מצומת מסוים מסלול לכל צומת אחר ומכל צומת יש מסלול אליו, אז יש דרכו מסלול מכל צומת לכל צומת אחר. מסלול מצומת אחר אליו משתמש באותן קשתות כמו מסלול ממנו לאותו צומת אחר בגרף שכיוון קשתותיו הפוך. בכל שלב אנו בוחרים בקשת בעלת משקל מינימלי שמאפשרת קשירות בין שתי קבוצות של צמתים.

<u>שאלה 31</u> **(מבחינה של פרופ' נוגה אלון וד''ר איריס רוזנבלום)**

נתון גרף מכוון $G=(V, E)$ המיוצג ע"י רשימות שכנות עם פונקציית משקל חיובית $w : E \to R^+$ ונתון $s\in V$. בנוסף, נתונה פונקציה $f : V \to R$. תארו אלגוריתם יעיל ככל האפשר הבודק האם לכל $v\in V$ מתקיים $\delta(s,v)=f(v)$, כאשר $\delta(s,v)$ מסמן את המרחק הקצר ביותר מ-s ל-v בגרף.

<u>יעילות</u>: $O(|V|+|E|)$

<u>אלגוריתם</u>:

נעבור על כל הקשתות. עבור כל קשת $e_{i,j}$ נוודא ש $f(v_j) \le f(v_i)+w_{i,j}$. כמו-כן נוודא שעבור כל צומת j קיימת קשת $e_{i,j}$ כך ש $f(v_j)=f(v_i)+w_{i,j}$. נבדוק גם אם $f(s)=0$. אם"ם כל אלה יתקיימו אז אלה המרחקים המינימלים מצומת s.

<u>הסבר</u>:

אם קיים צומת k שעבורו $f(v_k)>\delta(s,v_k)$ אז במסלול הקצר ביותר לצומת v_k קיים צומת ראשון v_j שעבורו $f(v_j)>\delta(s,v_j)$. אם v_i הוא אביו של v_j במסלול אז דרכו נקבל ש $f(v_j)>f(v_i)+w_{i,j}$.

נניח שקיים צומת k כך שעבורו $f(v_k)<\delta(s,v_k)$: נסתכל על הצומת j שעבורו $f(v_j)$ מינימלי מבין הצמתים שלגביהם $f(v_r)-\delta(s,v_r)$ מקבל ערך מינימלי. עבור צומת זה לא קיים אף צומת i כך ש $f(v_j)=f(v_i)+w_{i,j}$ הסבר: עבור כל צומת i מתקיים לפחות אחד מהשניים $f(v_i)\ge f(v_j)$ או $f(v_j)-f(v_i)<\delta(s,v_j)-\delta(s,v_i)\le w_{i,j}$.

אם הפונקציה f היא פונקצית מרחקים אז היא עונה לכל האילוצים. לכל צומת מתקבל מרחק מינימלי דרך אביו בעץ המסלולים המינימלים. על כל צומת ועל כל קשת אנו עוברים מספר קבוע של פעמים.

<u>שאלה 32</u> **(מבחינה של פרופ' נוגה אלון וד''ר איריס רוזנבלום)**

נתונה תבנית $P=P[1]P[2]\ldots P[m]$ מעל א"ב סופי $\Sigma\cup\{*\}$ שמכילה בדיוק הופעה אחת של הסימן * (למשל: ac*ba). כמו כן נתון טקסט $T=T[1]T[2]\ldots T[n]$ מעל הא"ב Σ. תארו אלגוריתם יעיל ככל האפשר שיקבע האם T מכיל מחרוזת כלשהיא שמתקבלת מ-P ע"י הצבת תת מחרוזת כלשהיא במקום * (למשל: הטקסט d**ac**da**ba**bd מכיל את התבנית ac*ba החל מהאות השניה).

<u>יעילות</u> : $O(n+m)$

<u>אלגוריתם והסבר</u>:

באמצעות אלגוריתם KMP נחפש את ההופעה הראשונה של תת המחרוזת שלפני * ונחפש את ההופעה המאוחרת ביותר של תת המחרוזת שאחרי *. נדרוש שהשניה תתחיל לאחר שהאחרונה תסתיים. אם * צריך להופיע במקום מחרוזת לא ריקה אז נדרוש שיהיה פער בין מקום הסיום של הראשונה ומקום ההתחלה של השניה. אם זה מתקיים אז נוכל להציב את * במקום מה שמפריד את הופעתם.

<u>שאלה 33</u> **(מבחינה של פרופ' נוגה אלון וד''ר איריס רוזנבלום)**

שני שחקנים נבונים, A ו-B, משחקים במשחק הבא: נתונה סדרה של n מספרים שלמים מסודרים בשורה משמאל לימין. השחקנים משחקים לסירוגין כשכל אחד בתורו לוקח את המספר הימני ביותר או את המספר השמאלי ביותר.השחקן A משחק ראשון. בסיום המשחק, הערך עבור השחקן A הוא סכום המספרים שהוא בחר פחות סכום המספרים ש-B בחר. הערך עבור השחקן B הוא, באופן דומה, סכום המספרים שהוא בחר פחות סכום המספרים ש-A בחר. מטרת כל שחקן להשיג ערך מירבי; לכן, למשל, אם הסדרה ההתחלתית 1,3 והשחקנים משחקים אופטימלית אזי הערך עבור A הוא 2 והערך עבור B הוא 2-, ואם הסדרה ההתחלתית היא 1, 8, 3 אזי הערך עבור A הוא 4- והערך עבור B הוא 4.

תארו אלגוריתם יעיל ככל האפשר שיחשב, בהינתן סדרה באורך n את ערך המשחק עבור השחקן A.

<u>יעילות</u>: $O(n^2)$

<u>אלגוריתם והסבר</u>:

יהיו $a_1, a_2, \cdots, a_n$ אברי הסדרה. נגדיר $f_{i,i} = a_i$ עבור כל $1 \le i \le n$. עבור כל $1 \le i < j \le n$ נגדיר $f_{i,j} = \max\{a_i - f_{i+1,j}, a_j - f_{i,j-1}\}$. שחקן יכול לקחת מאחד הקצוות שנותרו ואז התור עובר לאחר. עבור כל $i, j\ :\ 1 \le i < j \le n$ יש השוואה בין שני גדלים.

<u>שאלה 34</u> **(מבחינה של פרופ' נוגה אלון)**

נתון טקסט $T = T[1]T[2]\cdots T[n]$ ומחרוזת $P = P[1]P[2]\cdots P[m]$ מעל א"ב סופי $\sum$, ונניח כי כל התווים במחרוזת P שונים. תארו אלגוריתם יעיל ככל האפשר שימצא את כל ההסטים s,

$0 \le s \le n - m$, עבורם P מופיעה בהסט s ב- T עם לכל היותר 3 שיבושים, ז"א

$\left|\{i | 1 \le i \le m, P[i] \ne T[s+i]\}\right| \le 3$.

<u>סיבוכיות</u>: $O(n+m)$

<u>הסבר</u>:

נתחיל להשוות בין הטקסט למחרוזת החל מההתחלה עד מקום $k+1$ בו נגלה אי התאמה רביעית או התאמה מספקת (עם לכל היותר 3 שגיאות)לכל המחרוזת. אם $k < 7$ אז נתחיל תהליך זהה במקום אחד יותר מאוחר מהמקום שבו התחלנו פעם קודמת. אם $k \ge 7$ אז נתחיל ב $k - 6$ מקומות אחרי המקום שהתחלנו פעם קודמת.

לא תתכן חפיפה של יותר מ 6 תווים בין שתי רישות שונות ששתיהן מדויקות מספיק. בשתי רישות שמתאימות מספיק יש לכל היותר $2 \cdot 3 = 6$ מקומות שבהן התו בטקסט לא שווה לתו באחת הרישות. ביתר המקומות הוא צריך להיות שווה לתווים בשתי הרישות. אך כל התווים במחרוזת שונים והוא לא יכול להיות שווה לשני תווים שונים.

אנו משקיעים $k+1$ השוואות כדי להתקדם $k-6$ צעדים או לכל היותר 6 השוואות כדי להתקדם צעד אחד. לכן הסיבוכיות היא לינארית.

שאלה 35 (**מבחינה של פרופ' נוגה אלון וד''ר איריס רוזנבלום ומבחינה של פרופ' מיכה שריר**)

נתונה מטריצה n על n שאיבריה הם 0 או 1. אלכסון מוכלל של המטריצה הוא של n אחדות כך שמכל שורה ומכל עמודה נבחר אחד יחיד. לדוגמא , במטריצה משמאל מודגש אלכסון מוכלל. שימו לב שלא בהכרח קיים אלכסון מוכלל למטריצה.

0	1^*	0	1
0	0	1^*	1
1^*	0	1	0
0	1	0	1^*

להלן אלגוריתם לחישוב אלכסון מוכלל בהינתן המטריצה:

1. נחזיק מערך בוליאני בגודל n שבו נסמן כל טור שכבר "תפסנו" (בחרנו בו כבר '1 '). בהתחלה כל הטורים אינם תפוסים.
2. נעבור על המטריצה שורה-שורה:
   א. בכל שורה נחפש את ה-1 הראשון בטור שעוד לא תפסנו.
   ב. אם נמצא 1 כזה, נוסיף אותו לאלכסון המוכלל ונסמן שהטור תפוס.
   ג. אם לא נמצא 1 כזה בשורה הנוכחית-נעצור ונודיע שאין פתרון.

א) הוכיחו כי האלגוריתם שגוי.

נראה דוגמא שבה האלגוריתם לא נותן פתרון, למרות שיש פתרון.

1	1	0
1	0	0
0	0	1

אם נבחר את המשבצת השמאלית העליונה אז כבר לא יהיה פתרון. אבל יש פתרון כאשר נבחרת המשבצת האמצעית העליונה. אפשר גם לתת דוגמא עם מטריצה 2×2 .

ב) תארו אלגוריתם יעיל ככל האפשר שימצא אלכסון מוכלל של המטריצה, אם קיים, או יאמר בביטחון שאין.

__יעילות__: של חישוב זיווג מכסימלי בגרף דו צדדי ועוד מעבר על מטריצה רבועית

__אלגוריתם והסבר__:

נגדיר גרף דו צדדי שבו הצמתים מצד אחד ייצגו את השורות והצמתים מהצד השני ייצגו את העמודות.
תהיה קשת מצומת i לצומת j אם"ם במטריצה במשבצת ה- i, j מופיע 1.

בגרף זה נחשב זיווג מכסימלי. קיים פתרון אם"ם בגרף זה קיים זיווג מושלם. בזיווג מושלם, בכל צומת נוגעת קשת אחת. את בעיית הזיווג אפשר לפתור באמצעות פתרון בעיית זרימה. נוסיף מקור שממנו יצאו קשתות בקיבול 1 כל אחת לכל צמתי צד אחד ונוסיף יעד כך שמכל אחד מצמתי הצד השני יצאו קשתות אל היעד. לכל הקשתות יהיה קיבול 1 בכיוון מהצד הראשון לשני. ננסה להזרים n יחידות.

<u>שאלה 36</u> **(מבחינה של פרופ' נוגה אלון וד''ר איריס רוזנבלום ומבחינה של פרופ' מיכה שריר)**
נתון גרף אציקלי מכוון $G=(V,E)$ המיוצג ע"י רשימות שכנות ונתון זוג צמתים $x,y\in V$. תארו אלגוריתם יעיל ככל האפשר שיחשב את מספר המסלולים המכוונים בגרף G העוברים גם דרך x וגם דרך y.

<u>יעילות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:
נחשב מיון טופולוגי בגרף. נניח בלי הגבלת הכלליות שבמיון מופיע x לפני y.
נוסיף צומת s ונחברו בקשתות לכל הצמתים שמופיעים במיון הטופולוגי לא אחרי x. נוסיף צומת t ונחבר קשתות אליו מכל הצמתים שמופיעים במיון הטופולוגי לא לפני y.
נחשב את מספר המסלולים מ s ל x, מ x ל y ומ y ל t. הפתרון יהיה המכפלה של שלושת המספרים האלה. בחישוב כל אחד ממספרים אלה, נסתמך על המיון הטופולוגי ונשתמש בתכנות דינמי. למשל בחישוב מספר המסלולים מ x ל y, לגבי צומת w שבינהם כולל אותם, מספר המסלולים מ w ל y יהיה שווה לסכום מספרי המסלולים ל y מצמתים שיש מ w קשת אליהם (ברור שמסלול חייב לעבור דרך אחד מהם או שהוא ישיר ל y - שזהו מקרה פרטי).

<u>שאלה 37</u> **(מבחינה של פרופ' נוגה אלון, פרופ' עמוס פיאט, פרופ' רון שמיר ופרופ' מיכה שריר)**
נתון גרף פשוט, קשיר ולא מכוון $G=(V,E)$ עם פונקצית משקל שלמה על הקשתות $w:E\to Z$.
בנוסף ידוע כי כל המשקלים שונים, חוץ משתי קשתות e_1,e_2 עבורן מתקיים $w(e_1)=w(e_2)$.
תארו אלגוריתם יעיל ככל האפשר המוצא את <u>מספר</u> העצים הפורשים המינימאלים של G.

<u>יעילות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:
לגבי כל אחת משתי הקשתות e_1,e_2 נמצא אם היא נמצאת על מעגל שכולל חוץ ממנה רק קשתות קלות ממנה, ואם היא נמצאת על מעגל שכולל רק קשתות בעלות משקל יותר קל או שווה לשלה (הבדיקות יעשו על-ידי כך שנחפש מסלולים בין קצוות כל אחת מהקשתות).
אם לפחות אחת משתי הקשתות נמצאות על מעגל שכולל רק קשתות קלות מהן, אז יש עץ פורש מינימאלי יחיד (על פי נכונות אלגוריתם קרוסקל, כל קשת שאינה שנמצאת על מעגל שבו כל הקשתות קלות ממנה, לא נמצאת באף עץ פורש מינימאלי). אם לפחות אחת משתי הקשתות לא נמצאת על מעגל שבו כל הקשתות קלות או שוות משקל לה, אז יש עץ פורש מינימאלי יחיד. אם הן נמצאות על מעגל שבו כל הקשתות קלות או שוות למשקלן, אבל לא נמצאות על מעגל שבו כל הקשתות קלות מהן, אז יש שני עצים פורשים מינימאלים (אחרי שבוחרים את אחת מהן אי אפשר לבחור את האחרת לאותו עץ פורש מינימאלי).

<u>שאלה 38</u> **(מבחינה של פרופ' מיכה שריר ודן פלדמן)**

יהא $G=(V,E)$ גרף לא מכוון מיוצג ע"י רשימות שכנות, עם משקל שלם $w(e)$ לכל קשת $e\in E$.
לכל מספר ממשי x, נסמן ב- $G^{(x)}=\left(V,E^{(x)}\right)$ את תת הגרף המורכב מכל הקשתות שמשקלן לכל היותר x. תארו אלגוריתם יעיל ככל האפשר המחשב את ה- x המקסימלי עבורו $G^{(x)}$ חסר מעגלים.

<u>יעילות</u>: $O(\min(|V|,|E|)\log(|E|)+|V|+|E|)$

<u>אלגוריתם</u>:

נזרוק מהגרף את כל הקשתות שהן לא בין $|V|$ הקשתות הקלות ביותר.
נבצע את השלב הראשון של האלגוריתם של קרוסקל (מיון הקשתות לפי משקלן).
נבצע את השלב השני ונקטע אותו כאשר תמצא קשת שסוגרת מעגל עם קשתות שלא כבדות ממנה.
עבור כל מספר x שקטן ממש ממשקל קשת זו, $G^{(x)}$ הוא חסר מעגלים.

<u>הסבר</u>:

כל עוד לא התגלתה קשת שסוגרת מעגל אז אפשר לבחור את כל הקשתות מבלי שיווצר מעגל.
בכל קבוצה של $|V|$ קשתות יש בודאי קשת שיוצרת מעגל עם קשתות שאינן כבדות ממנה.
אפשר למצוא את הקשתות שצריך למחוק בזמן לינארי (סטטיסטי סדר).

<u>שאלה 39</u> **(מבחינה של פרופ' מיכה שריר ודן פלדמן)**

נתון גרף מכוון $G=(V,E)$ על ידי רשימות שכנויות, ומשקל $w(e)$ חיובי לכל $e\in E$. הגרף מייצג רשת תקשורת, כאשר כל צומת מייצג משתמש, וכל קשת מייצגת קשר בין שני משתמשים. משתמש u יכול לשלוח הודעה למשתמש v אם קיים מסלול (מכוון) מ- u ל- v בגרף. משקל הקשת $(u,v)\in E$ הוא עלות הניתוק של הקשר הישיר בין המשתמשים u ו- v. כל צומת $v\in V$ מסווג כ"בוגר", "קטין", או "לא ידוע".
מטרתנו היא לחשב תת-קבוצה E' של קשתות הגרף כך שאם נמחק את כל קשתות E' מהגרף, לא יהיה מסלול משום צומת "קטין" לשום צומת "בוגר". תארו אלגוריתם יעיל ככל האפשר המקבל כקלט את w,G, ואת סיווג הצמתים, ומחזיר קבוצת קשתות E' המקיימת את התכונה לעיל, בעלת משקל כולל מינימלי מבין כל הקבוצות המקיימות את התכונה.

<u>יעילות</u>: של אלגוריתם לחישוב זרימה מקסימלית

<u>אלגוריתם</u>:

נוסיף מקור ויעד. נוסיף קשתות מהמקור לכל צומת "קטין" ומכל צומת "בוגר" ליעד. נתן קיבולים לכל קשתות הגרף. הקיבול של כל קשת מקורית יהיה שווה למשקלה והקיבול של כל קשת שהוספה יהיה שווה לסכום כל קיבולי הקשתות המקוריות. נפעיל אלגוריתם לחישוב זרימה מקסימלית מהמקור ליעד. אלגוריתם זה ימצא חתך מינימלי. הקשתות שנחזיר הן קשתות החתך.

<u>הסבר</u>:

יש להסיר את כל הקשתות באיזשהו חתך כדי לנתק את הקשר. מכיון שלקשתות החדשות נתנו קיבול גדול אז החתך יהיה בין צמתים מהגרף המקורי.
<u>הערה</u>: את הקשתות המחברות "קטין" ל"בוגר" יכולנו מראש להסיר. יש גרפים בהם זה משנה את סדר הגודל של היעילות (כאשר מספר הקשתות הנוגעות ב"לא ידועים" הוא קטן).

<u>שאלה 40</u> **(מבחינה של פרופ' מיכה שריר ודן פלדמן)**

נתונה קבוצה E של n האינטרוולים על הישר $[a_1,b_1],\cdots,[a_n,b_n]$ כשלכל אינטרוול $[a_i,b_i]$ יש משקל חיובי w_i. תארו אלגוריתם יעיל ככל האפשר שמחשב תת-קבוצה E' של E עם משקל מקסימלי כך שכל האינטרוולים ב- E' זרים בזוגות (החיתוך של כל שניים ריק). (**<u>רמז</u>**: סדרו את האינטרוולים בסדר עולה של נקודות הקצה הימניות שלהם, וקבלו נוסחת נסיגה עבור $OPT(j)$, האופטימום עבור j האינטרוולים הראשונים.)

<u>יעילות</u>: $O(n\ln(n))$

<u>אלגוריתם</u>:

<u>שלב ראשון</u>: נמיין את האינטרוולים לפי סדר עולה של הקצה הימני שלהם.

<u>שלב שני</u>: עבור כל אינטרוול j נמצא באמצאות חיפוש בינארי מבין האינטרוולים שקדמו לו במיון את האינטרוול $m(j)$ בעל הקצה הימני ביותר שאיתו הוא לא נחתך.

<u>שלב שלישי</u>: נעבור על פני כל האינטרוולים על פי הסדר שנקבע בשלב הראשון. עבור כל j נגדיר

$$OPT(j)=Max\{OPT(j-1),OPT(m(j))+w_j\}$$

$$OPT(0)=0$$

<u>הסבר</u>:

בכל שלב אנו מקבלים את הפתרון הטוב ביותר שלא כולל את האינטרוול הנבדק או שכולל אותו וגם קבוצה מקסימלית מבין האינטרוולים הקודמים שאינם נחתכים איתו.

<u>שאלה 41</u> **(מבחינה של פרופ' מיכה שריר ודן פלדמן)**

נתון גרף מכוון $G=(V,E)$ המיוצג ע"י רשימות שכנות, ונתונה תת-קבוצה $E'\subseteq E$ של קשתות "חשובות".

תארו אלגוריתם יעיל ככל האפשר, הבודק אם קיים ב- G מסלול העובר דרך כל הקשתות של E'.

<u>יעילות</u>: $O(|V|+|E|)$

<u>אלגוריתם</u>:

בגרף הנתון נוסיף עבור כל קשת חשובה צומת ונפצל את הקשת לשתי קשתות. הראשונה תחבר את צומת המקור לצומת הנוסף של אותה קשת והשניה את הצומת הנוסף לצומת היעד של הקשת. נריץ את האלגוריתם שתואר בתשובה לשאלה 21 בקובץ זה לבדיקה האם הצמתים שהוספו נמצאים על מסלול אחד

<u>הסבר</u>:

יש מסלול העובר בכל הצמתים שהוספו אם"ם יש בגרף המקורי מסלול שעובר בכל הקשתות המיוחדות. לא הגדלנו את סדר הגודל של סכום מספר הצמתים והקשתות בגרף.

<u>הערה</u>: אם נדרוש שהמסלול יהיה פשוט אז הבעיה היא NPC.

<u>שאלה 42</u> **(מרצים: פרופ' נוגה אלון , ד"ר איריס רוזנבלום , מתרגלים: רני הוד , דני פלדמן)**
נתונה סדרה $X = x_1 x_2 \cdots x_n$ כאשר לכל $1 \le i \le n$, $x_i \in \{a,b,c\}$. תאר/י אלגוריתם יעיל ככל האפשר המחשב רצף $x_i x_{i+1} \cdots x_{j-1} x_j$ עבורו מספר ההופעות של a פחות מספר ההופעות של b מקסימלי. במילים אחרות,

$$\left|\{p : i \le p \le j, x_p = a\}\right| - \left|\{p : i \le p \le j, x_p = b\}\right|$$

מקסימלי מבין הבחירות האפשריות ל- i, j עבור $1 \le i \le j \le n$.

<u>יעילות</u>: $O(n)$

<u>אלגוריתם והסבר</u>:
בעזרת תכנות דינמי נחשב את ערכי F_p שמייצגים את הפתרון הטוב ביותר שמתחיל בנקודה p.

$$F_n = \begin{cases} 0 & x_n \in \{b,c\} \\ 1 & x_n = a \end{cases}$$

ועבור כל $1 \le p \le n-1$:

$$F_p = \begin{cases} F_{p+1} + 1 & x_p = a \\ F_{p+1} & x_p = c \\ \max\{F_{p+1} - 1, 0\} & x_p = b \end{cases}$$

אחר-כך נעבור על פני כל ערכי F_p. הרצף המבוקש יתחיל בנקודה שבה F_p הוא מקסימלי ויסתיים בנקודה המוקדמת ביותר אחר-כך שבה הוא אפס, או ימשך עד הסוף אם עד הסוף אין נקודה שבה הפונקציה מתאפסת.
אם מנקודה מסוימת אי אפשר ליצור רצף שבו יותר a מ b, אז כדאי לשמוט את הקרן שמתחילה בנקודה זו. אחרת כדאי לצרף את הרצף הזה לפתרון שמגיע עד אליה.

<u>שאלה 43</u> **(מרצים: פרופ' נוגה אלון , ד"ר איריס רוזנבלום , מתרגלים: רני הוד , דני פלדמן)**
נתונה רשת זרימה $G = (V, E)$ עם מקור s, בור t ופונ' קיבול $c : E \to R^+$ ונתונה זרימה f ברשת. נניח שמצאנו מסילה משפרת $P : s \rightsquigarrow t$ ברשת השיורית G_f שאורכה בקשתות מינימלי. תהיה g הזרימה ברשת G המתקבלת מ- f לאחר שהזרמנו לאורך P את הקיבול השיורי $c(P)$.
הוכח/הוכיחי כי לכל צומת $v \in V$ מתקיים $\delta_g(s,v) \ge \delta_f(s,v)$ כאשר δ_f, δ_g הם המרחקים ברשתות השיוריות G_f, G_g בהתאמה.

<u>הוכחה</u>:
נניח שיש צמתים שעבורם זה לא מתקיים. אז לאחר העדכון קיים צומת v שהוא הראשון במסלול שמוביל אליו שעבורו המסלול החדש קצר יותר. לאחר העדכון מגיעים אליו מצומת u שאליו כעת המסלול קצר יותר ב 1 ולא יותר קצר מאשר היה קודם. קודם לא יכולנו להגיע ל v דרך u (כי אחרת המסלול ל v לא היה ארוך יותר קודם). לכן נוכל להסיק שבעדכון הזרמנו מ v ל u (כך שהתאפשרה אחר-כך זרימה בכיוון ההפוך). מכיון שהמרחק ל v היה קודם ארוך יותר מאשר עכשיו ומ v עברנו ל u אז המרחק ל u היה קודם ארוך יותר מאשר עכשיו ל v ובודאי גם מהמרחק עכשיו ל u.
זאת סתירה לכך שקיימים צמתים שהמרחק אליהם התקצר.

<u>שאלה 44</u> **(מרצים: פרופ' נוגה אלון , ד''ר איריס רוזנבלום , מתרגלים: רני הוד , דני פלדמן)**
נתון גרף $G=(V,E)$ מכוון וקשיר בחוזקה המיוצג ע"י רשימות שכנות, נתונה פונקצית משקל $w: E \to R$ ונתונים זוג צמתים שונים $u,v \in V$. ידוע שאין בגרף מעגל שלילי ביחס ל- w.
תאר/י אלגוריתם יעיל ככל האפשר שימצא, עבור כל ערך של k, $2 \le k \le |V|-1$, את המשקל הקל ביותר של מסילה מכוונת $u \rightsquigarrow v$ שאורכה לכל היותר k קשתות (אם יש כזו).
הערה: האלגוריתם מחשב, אם כן, $|V|-2$ ערכים מ- $R \cup \{\infty\}$.

<u>יעילות</u>: $O(|V||E|)$

<u>אלגוריתם והסבר</u>:
נבנה גרף בעל $|V|$ שכבות של צמתים. בשכבה הראשונה יהיה רק הצומת u. בכל אחת מהשכבות האחרות יהיה עותק של כל צומת מצמתי הגרף. מהשכבה הראשונה לשכבה השניה יצאו כל הקשתות שיוצאות מהצומת u בגרף המקורי. מצומת שבשכבה אחרת תצא קשת לנציג של צומת אחר שבשכבה הבאה אם"ם יש ביניהם קשת בגרף המקורי. כמו-כן תצא קשת מכל נציג לנציג של אותו צומת שבשכבה הבאה. משקלי הקשתות שבין נציגים של צמתים שונים יהיו תמיד זהים למשקלם בגרף המקורי ומשקלי הקשתות שבין נציגים של אותו צומת יהיו תמיד שווים לאפס. בהינתן המרחק לצמתים שבשכבה מסוימת, יהיה המרחק לכל צומת שבשכבה הבאה שווה למינימום על-פני הצמתים שבשכבה הקודמת, שמהם נכנסת לצומת קשת, של המשקל אליהם ועוד משקל הקשת לצומת שבשכבה הבאה. המשקלים המבוקשים יהיו המשקלים לנציגי הצומת v שבשכבות השונות. בהינתן שחישבנו את המשקלים לשכבה קודמת אז המשקלים לשכבה הבאה יכולים להשתמש בקשת אחת יותר או לא להשתמש ביותר קשתות ואז משתמשים בקשת בעלת משקל של אפס. אם עד שכבה מסוימת אין מסלול אז המשקל של מסלולים בעלי אורך לכל היותר זה הוא ∞. מכיון שהגרף הוא קשיר בחוזקה אז מספר הקשתות בו אינו קטן ממספר צמתיו. עבור כל זוג שכבות שכנות יהיה לנו סדר גודל של $|E|$ השוואות.

<u>שאלה 45</u> **(מרצים: פרופ' נוגה אלון , ד''ר איריס רוזנבלום , מתרגלים: רני הוד , דני פלדמן)**
בבית הספר היסודי "סדר מעל לכל" לומדים תלמידים בכיתות א'-ו'. בכל כיתה לומדים n תלמידים.
במסדר הסיום רוצה המנהל לסדר את התלמידים ב־n טורים כשבראש כל טור ילד מכיתה א', אחריו ילד מכיתה ב', בעקבותיו ילד מכיתה ג', וכך הלאה עד לסוף הטור שבו ילד מכיתה ו'. לכל ילד מכיתה ב' יש רשימה של ילדים מכיתה א' שהוא מוכן לעמוד בעקבותיהם בטור. באופן דומה, לכל ילד מכיתה ג' יש רשימה של ילדים מכיתה ב' שהוא מוכן לעמוד בעקבותיהם בטור, וכך הלאה עד לילדי כיתה ו' שלכל אחד מהם רשימה של ילדים מכיתה ה' שהוא מוכן לעמוד בעקבותיהם.
המנהל מטיל על המורה לאלגוריתמים לבדוק האם אפשר לסדר את $6n$ הילדים ב־n טורים תוך שמירה על אוסף האילוצים הנתון, תאר/י אלגוריתם יעיל ככל האפשר בו תשתמש המורה כדי לפתור את הבעיה.

<u>יעילות</u>: של אלגוריתם לחישוב זיווג מכסימלי בגרף דו-צדדי

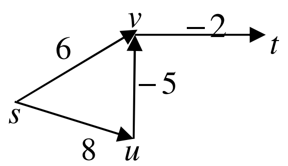
<u>אלגוריתם והסבר</u>:
נפתור 5 בעיות זיווג. עבור כל i: $2 \le i \le 6$ נחשב זיווג מכסימלי בגרף הדו-צדדי ששתי קבוצות צמתיו מייצגות את ילדי כיתה i ואת ילדי כיתה $i-1$. בין כל זוג צמתים כאלה יש קשת אם"ם הילד מכיתה i מוכן לעמוד מאחורי הילד מכיתה $i-1$. אם"ם בכל 5 הבעיות יש זיווג מושלם אז יש פתרון כנדרש.

<u>שאלה 46</u> (מרצים: פרופ' נוגה אלון , ד"ר איריס רוזנבלום , מתרגלים: רני הוד , דני פלדמן)

<u>סעיף א'</u>

תאר/י דוגמא של גרף $G=(V,E)$ עם פונ' משקל $w:E \to R$ וצומת $s \in V$ כך שאין מעגל שלילי ביחס ל־ w בגרף והאלגוריתם של Dijkstra נכשל; במילים אחרות, הראה/הראי צומת $v \in V$ כך שהאלגוריתם <u>אינו</u> מחשב נכון את אורך המסילה הקלה ביותר מ־ s ל־ v.

<u>דוגמא והסבר</u>:



הקשת מ־ s ל־ v היא הקשת הקלה ביותר שיוצאת מ־ s. לכן בשלב הראשון יקבע מרחק 6 מ־ s ל־ v. אך דרך u יש מסלול מ־ s ל־ v שאורכו $8-5=3$. המסלול ל־ t עובר דרך v. אבל הוא כבר אף פעם לא יעודכן על סמך המשקל הנכון ל־ v.

<u>סעיף ב'</u>

נתון גרף לא מכוון $G=(V,E)$ המיוצג ע"י רשימות שכנות עם פונ' משקל חיובית $w:E \to R^+$ ונתון צומת $s \in V$. ידוע שדרגת כל צומת בגרף G היא לכל היותר 10.

תאר/י אלגוריתם יעיל ככל האפשר שימצא את 100 הצמתים הקרובים ביותר ל־ s. [1]

[1] במקרה של שיוויון בין מרחקים נעדיף צמתים עם מס' סידורי קטן יותר.

<u>יעילות</u>: $O(1)$

<u>אלגוריתם והסבר</u>:

נריץ את אלגוריתם דיקסטרה עד הוספת 100 צמתים לקבוצת הצמתים שאת המרחק הסופי אליהם חישבנו. בכל שלב נחשב את המרחק לכל השכנים של הצמתים שכבר בקבוצה. מרחק זה יהיה שווה למינימום על פני הסכום של אורכי המסלולים לצומת שכבר בקבוצה + אורך הקשת מהצומת הזה לצומת שעדיין לא בקבוצה. בכל שלב נוסיף לקבוצה את הצומת בעל האינדקס המינימלי מבין הצמתים הקרובים ביותר שעדיין לא היה בקבוצה. המרחק הזה אליו יהיה סופי. בכל שלב לא עוברים בבדיקה על יותר מ $100*10=1000$ קשתות.

<u>הערה</u>: כדי להגיע לסיבוכיות זו, לא נזקקנו למבנה הנתונים שבו משתמשים בדרך כלל באלגוריתם דיקסטרה.

<u>שאלה 47</u> (מרצים: פרופ' יוסי עזר , פרופ' רון שמיר , מתרגלים: ידעאל ולדמן , אדם שפר)

סעיף א'

נתונים רשת זרימה מכוונת $G=(V,E)$ וזוג קודקודים $s,t \in V$, כך שבנוסף לפונקציית הקיבול על הקשתות $C:E \to N_1$, קיימת גם פונקציית קיבול על הצמתים $D:V \to N_1$ (תזכורת: N_1 הינה קבוצת השלמים החיובים). נגיד שזרימה היא חוקית אם היא מקיימת את אילוצי הזרימה הרגילים,

ובנוסף- הזרימה שנכנסת לכל צומת אינה עולה על קיבול הצומת (במקרה של המקור, הזרימה היוצאת מהצומת).
תארו אלגוריתם יעיל למציאת זרימה חוקית מקסימלית מ־s אל t.

<u>יעילות</u>: של אלגוריתם לזרימה מכסימלית

<u>אלגוריתם והסבר</u>:
נפצל כל צומת חוץ מ s ו t לשניים. הקשתות הנכנסות לצומת המקורי יכנסו לצומת הראשון והקשתות היוצאות יצאו מהצומת השני. תהיה קשת מהצומת הראשון לצומת השני. הקיבול של קשת זו יהיה שווה לקיבול הצומת המקורי. נמצא זרימה מכסימלית מ s ל t. כל זרימה אפשרית עומדת באילוצי הקיבולת. לא שינינו את סדר הגודל של מספר הצמתים והקשתות.

סעיף ב'
כיצד תשתנה התשובה לסעיף א' אם נתון שכל קיבולי הקודקודים, פרט לאלו של s ושל t, הם 1 ?

<u>יעילות</u>: של אלגוריתם לחישוב זרימה מכסימלית ברשת מטיפוס 2

<u>אלגוריתם והסבר</u>:
מכיון שקיבולי הצמתים הם 1 אז לא ניתן להזרים באף קשת יותר מיחידה אחת. לכן נוכל להוריד את קיבולי כל הקשתות ל 1. נמצא זרימה מכסימלית מ s ל t. ברשת זו יש לכל צומת דרגת כניסה או יציאה של 1.

<u>שאלה 48</u> **(מרצים: פרופ' יוסי עזר , פרופ' רון שמיר , מתרגלים: ידעאל ולדמן , אדם שפר)**
נתון גרף קשיר ולא מכוון $G=(V,E)$ עם פונקציית משקל חיובית על הקשתות $w:E\rightarrow(0,\infty)$.
בנוסף, נתונה קבוצת קשתות $E^{'}\subseteq E$ כך שתת הגרף $G^{'}=(V,E^{'})$ חסר מעגלים. תארו אלגוריתם יעיל אשר מוצא, מבין העצים הפורשים המכילים את כל קשתות $E^{'}$, עץ בעל משקל מינימאלי (שימו לב שלא בהכרח מדובר בעפ"מ).

<u>יעילות</u>: $O\left(|E|+\left(|V|-\left|E^{'}\right|\right)\log\left(|V|-\left|E^{'}\right|\right)\right)$
(מכיון שצריך למצוא עץ פורש שמכיל קבוצה בת $\left|E^{'}\right|$ קשתות אז ניתן להניח ש $|V|>\left|E^{'}\right|$).

<u>אלגוריתם והסבר</u>:
בשלב הראשון נמצא את רכיבי הקשירות של הגרף $G^{'}=(V,E^{'})$ (נעשה זאת על-ידי סדרת תהליכי BFS). כל רכיב קשירות יאוחד לצומת בודד שלו יהיו כל הקשתות של כל צמתי הרכיב (אם יש קשתות מקבילות אז אפשר להשאיר רק את הנמוכה שבהן). בגרף שמתקבל נמצא עץ פורש מינימלי על-ידי שימוש באלגוריתם של פרים.
חייבים להשתמש בכל קשתות $E^{'}$. כך נוצרים רכיבי קשירות. האלגוריתם של פרים בוחר בכל שלב קשת שהיא הקלה ביותר מבין הקשתות שבין שתי קבוצות זרות של צמתים. אם היא לא היתה מופיעה בפתרון אז יכולנו להוסיף אותה לפתרון והיה נוצר מעגל שבו לפחות קשת אחת לא יותר קלה. יכולנו למחוק את הקשת הלא יותר קלה ולקבל פתרון לא פחות טוב.

<u>שאלה 49</u> **(מרצים: פרופ' יוסי עזר , פרופ' רון שמיר , מתרגלים: ידעאל ולדמן , אדם שפר)**

סעיף א'

נתונים גרף מכוון $G=(V,E)$, זוג קודקודים $s,t\in V$, ופונקציית משקל $w:E\to R$. בנוסף, ידוע שאין בגרף מעגלים שליליים. תארו אלגוריתם יעיל המוצא מבין המסלולים הקצרים ביותר (ביחס למספר הקשתות) מ־s ל־t מסלול קל ביותר ביחס ל־w.

<u>יעילות</u>: $O(|E|+|V|)$

<u>אלגוריתם והסבר</u>:

נבצע BFS החל מצומת s. לגבי כל צומת שאליו כבר הגענו נשמור גם את המרחק לפי w במסלול הקצר ביותר אליו לפי ספירת קשתות. כשמגיעים לצומת אז כבר ידועים המסלולים האופטימליים לכל הצמתים שבשכבה שלפניו. כשנגיע לצומת שעדיין לא הגענו אליו אז נעדכן את המסלול והמרחק אליו בספירת קשתות (שקודם היה ∞) וגם את המרחק לפי w. כאשר מגיעים לצומת שאליו כבר הגענו במסלול בעל אותו אורך אז נעדכן את המרחקים אליו אם יש שיפור לפי w תוך שימוש בקשת האחרונה. בכל המקרים האחרים לא נעדכן את המסלול.

<u>הערה לגבי דרך אחרת</u>

ניתן למצוא את הגרף המכוון וחסר המעגלים של המסלולים הקצרים ביותר מ s אל יתר הצמתים בספירת קשתות. על העץ הזה ניתן בזמן לינארי למצוא עבור כל הצמתים את המסלולים הקלים ביותר לפי w.

סעיף ב'

נתונים גרף מכוון $G=(V,E)$, זוג קודקודים $s,t\in V$ ושתי פונקציות משקל $w,w':E\to R$. בנוסף, ידוע שאין בגרף מעגלים שליליים, גם לפי משקלי w וגם לפי משקלי w'. תארו אלגוריתם יעיל אשר מוצא, מבין המסלולים הקלים ביותר מ־s אל t לפי פונקציית המשקל w, מסלול קל ביותר מ־s אל t לפי פונקציית המשקל w'.

<u>יעילות</u>: $O(|V||E|)$

<u>אלגוריתם והסבר</u>:

נבנה גרף של $|V|$ שכבות. בשכבה הראשונה יהיה רק נציג של s ובשכבה האחרונה יהיה רק נציג של t. ביתר השכבות יהיו נציגים של כל הצמתים. בין כל שני נציגים של שני צמתים שהם בשכבות עוקבות תהיה קשת אם בגרף המקורי יש ביניהם קשת. כמו כן תהיה קשת בין כל שני נציגים של t. לא יהיו קשתות בין שכבות שאינן עוקבות. נשתמש בפונקציות המשקל המקוריות ונתן לקשתות שבין שני נציגים של t משקל 0. לאחר שחישבנו את המרחקים לצמתי שכבה i, נעבור לחישוב המרחקים לצמתי שכבה $i+1$. נעבור על כל הקשתות שבין שתי השכבות. כאשר נעבור על קשת מסוימת אז נעדכן את המסלול לפי w אם יש ממש שיפור לפי w כאשר משתמשים בקשת זו או שמתקבל אותו אורך כמו הקיים לפי w ויש שיפור לפי w'. כאשר קובעים מסלול מתוקן לצומת בשכבה מסוימת אז נעדכן גם את המרחק לפי w' (שאינו בהכרח אופטימלי לפי w'). המסלול ל t משתמש בלכל היותר $|V-1|$ מעברים ואז נשאר ב t.

<u>שאלה 50</u> **(מרצים: פרופ' יוסי עזר , פרופ' רון שמיר , מתרגלים: ידעאל ולדמן , אדם שפר)**

נתון גרף מכוון $G=(V,E)$ וקבוע שלם k. תארו אלגוריתם יעיל למציאת תת-קבוצה $U\subseteq V$ בגודל k אשר מקיימת את התכונות הבאות. לכל קודקוד $v\in V\setminus U$ קיים קודקוד $u\in U$ כך שישנו מסלול מכוון מ־v אל u, ובנוסף לא קיים מסלול מכוון בין שני איברים שונים של U. אם לא קיימת תת-קבוצה מתאימה, על האלגוריתם להתריע על כך.

<u>יעילות</u>: $O(|E|+|V|)$

<u>אלגוריתם והסבר</u>:

באמצעות שתי הרצות של DFS נמצא את רכיבי הקשירות החזקה.
מכל רכיב קשירות חזקה שממנו אין יציאות לרכיבי קשירות אחרים, ניקח נציג שרירותי.
אם מספר הנציגים שנבחרו הוא k אז קבוצת הנציגים שנבחרו היא פתרון.
אם מספרם הוא אחר אז אין פתרון.
בכל רכיב קשירות חזקה יש מסלולים בשני הכיוונים בין כל שני צמתים. לכן אי אפשר לבחור יותר מנציג אחד מאותו רכיב קשירות חזקה. אי אפשר לבחור בצומת מרכיב קשירות חזקה שממנו יש מסלול לרכיב קשירות אחר. לא משנה באיזה נציג בוחרים מרכיב שממנו אין מסלולים.

<u>שאלה 51</u> **(מרצים: פרופ' יוסי עזר , פרופ' רון שמיר , מתרגלים: ידעאל ולדמן , אדם שפר)**

נתון גרף קשיר, לא מכוון וממושקל $G=(V,E)$. תארו אלגוריתם יעיל אשר בודק האם ל-G יש עפ"מ יחיד. מספיק למצוא אלגוריתם שסיבוכיות הזמן שלו הינה $O(|E|\log|V|)$.

<u>יעילות</u>: של האלגוריתם של פרים

<u>אלגוריתם והסבר</u>:

נמצא עץ פורש מינימלי.
כעת נמצא שוב עץ פורש מינימלי בעזרת האלגוריתם של פרים, כאשר כרגיל נתן עדיפות לכל קשת על פני קשתות כבדות ממנה, אך נתן גם עדיפות לכל קשת שלא היתה בעץ הראשון על פני קשת בעלת משקל שווה לה שהיתה בעץ הראשון. אם"ם באיזשהו שלב תבחר לעץ קשת שלא היתה בעץ הראשון אז נדע שיש יותר מעץ פורש מינימלי יחיד. על-פי הנכונות של האלגוריתם של פרים, מותר היה לבחור בכל קשת כזאת. אם לא נבחרה אף קשת כזאת אז כל קשתות העץ המקורי מפרידות בין חלקי גרף שכל הקשתות שבינהם הן כבדות יותר.

<u>שאלה 52</u> **(מרצים: פרופ' נוגה אלון , פרופ' יוסי עזר , פרופ' רון שמיר , פרופ' מיכה שריר , מתרגלים: רני הוד , ידעאל ולדמן , אדם שפר)**

נתונים גרף מכוון $G=(V,E)$ עם משקל חיובי $w(e)$ עבור כל קשת $e\in E$, ומקור $s\in V$. תארו אלגוריתם יעיל ככל האפשר שימצא, לכל $v\in V$, את משקל המסילה המכוונת הקלה ביותר מ- s אל v שמספר קשתותיה זוגי וגם גדול או שווה ל-6 (המסילה אינה חייבת להיות פשוטה).

<u>יעילות</u>: של האלגוריתם של דיקסטרה

<u>אלגוריתם והסבר</u>:

נבנה גרף שבו 7 שכבות $i=1,2,3,4,5,6,7$. בכל שכבה יהיו נציגים של כל הצמתים של הגרף המקורי. יהיו קשתות רק בין שכבות שכנות. תהיה קשת מצומת i בשכבה מסוימת לצומת j בשכבה שכנה אם"ם בגרף המקורי יש קשת מצומת i לצומת j . נמצא מסלול בעל אורך מינימלי מצומת s שבשכבה הראשונה לצמתים שבשכבה 7.

מסלול כזה משתמש בלפחות 6 מקשתות הגרף המקורי ואורכו זוגי. לכל מסלול בעל אורך זוגי שאינו קצר מאורך 6 יש ייצוג בגרף זה. לא שינינו את סדרי הגודל של מספר הצמתים ומספר הקשתות.

<u>שאלה 53</u> **(מרצים: פרופ' עמוס פיאט , פרופ' מיכה שריר , מתרגלים: רני הוד , אדם שפר)**

נתונים גרף מכוון $G=(V,E)$, פונקצית קיבול אי-שלילית על הקשתות $c:E\to R^+$, ופונקצית מחיר אי-שלילית על הקשתות $p:E\to R^+$, ברצוננו לשנע m סוגים של מחצבים בעזרת הרשת. עבור $i=1,2,\cdots,m$, יש להעביר d_i ק"ג של מחצב i ממכרה הממוקם בצומת $s_i\in V$ למפעל הממוקם בצומת $t_i\in V$. עלות השימוש בקשת $e\in E$ היא $p(e)$ שילינג לכל ק"ג שעובר דרכה, ללא תלות בסוג המחצב; בנוסף, סך המשקל שיכול לעבור בקשת זו מוגבל ל-$c(e)$ ק"ג לכל היותר. מבין אסטרטגיות השינוע אשר מספקות את האילוצים לעיל, אנו מחפשים אחת שמביאה למינימום את העלות הכוללת, שהיא סכום עלויות הקשתות.

נסחו תוכנית לינארית לבעיה והסבירו אותה בקצרה.

<u>תוכנית לינארית והסבר</u>:

$x_{i,j,k}$ - כמות הזרימה של מוצר מסוג i שנזרים מצומת j לצומת k אם יש קשת מ j ל k.

פונקצית המטרה היא $\min\sum_{i,j,k}p_{j,k}x_{i,j,k}$ כאשר הסכימה היא על כל j,k שעבורם יש קשת מ j ל k.

האילוצים הם:

לכל i: $\sum_k x_{i,s_i,k}=d_i$, $\sum_j x_{i,j,t_i}=d_i$ ולכל i ולכל $j\neq s_i,t_i$: $\sum_k x_{i,k,j}=\sum_k x_{i,j,k}$ כאשר

לכל j : $x_{i,j,s_i}=0$ ו $x_{i,t_i,j}=0$ ויש אילוצי אי שליליות על כל המשתנים.

(כך מצומת המקור של כל מחצב יש יציאה של כמות d_i של מחצב זה , לצומת היעד יש כניסה של כמות זו ובכל צומת אחר, מה שנכנס הוא שיוצא).

ובנוסף לכל j,k שיש בינהם קשת יש אילוץ: $\sum_i x_{i,j,k}\le C_{j,k}$.

(אלה הם האילוצים על הזרימה הכוללת בכל קשת).

<u>שאלה 54</u> (מרצים: פרופ' עמוס פיאט , פרופ' מיכה שריר , מתרגלים: רני הוד , אדם שפר)
נתון גרף מכוון $G=(V,E)$ עם פונקצית משקל $\omega : E \to R$ ונתון צומת $s \in V$. ידוע שהקשת $e \in E$ היא הקשת היחידה בעלת משקל שלילי. תארו אלגוריתם בסיבוכיות $O(|E|+|V|\log|V|)$ אשר מוצא את המסלולים הקלים ביותר מ- s לכל שאר צמתי הגרף.

(מותר היה להניח שאין בגרף מעגלים בעלי אורך שלילי).

<u>אלגוריתם והסבר</u>:
הקשת e מחברת את w_1 ל w_2 נסתכל על הגרף המתקבל מהסרת קשת זו מהגרף המקורי. בגרף זה נחשב את המרחקים מצומת s לכל צמתי הגרף ומצומת w_2 לכל צמתי הגרף. עבור כל צומת בגרף, המסלול האופטימלי בגרף המקורי הוא המינימלי מבין המסלול אליו מצומת s שלא משתמש בקשת e והמסלול שמורכב מהמסלול מ s ל w_1 בצרוף הקשת e ובצרוף המסלול מ w_2 לצומת. כך יש חלוקה לשני מקרים שהוא משתמש או לא משתמש בקשת e. מכיון שאין מעגלים שליליים בגרף אז מסלולים אופטימלים לא משתמשים באף קשת כולל הקשת e יותר מפעם אחת. מחשבים מרחקים רק בגרף שמשקלי קשתותיו הם אי שליליים, לכן אפשר להשתמש באלגוריתם דיקסטרה.

<u>שאלה 55</u> (מרצים: פרופ' נוגה אלון, פרופ' יוסי עזר, פרופ' רון שמיר, מתרגלים: רני הוד, אדם שפר)
נתונה רשת זרימה מכוונת $G=(V,E)$ עם קיבולים <u>שלמים</u> $C : E \to N$, זוג צמתים $s,t \in V$ וזרימה מקסימלית שלמה[1] f מ s אל t. מגדילים ב 1 את הקיבול של שבע מקשתות הרשת $e_1,\cdots,e_7$, ומקטינים ב 1 את הקיבול של קשת שמינית $e_8 \in E$ (כל שמונה הקשתות שונות, והקיבול של e_8 ברשת המקורית חיובי). תארו אלגוריתם למציאת זרימה מקסימלית מ s אל t ברשת החדשה.

[1] תזכורת- זרימה שלמה היא זרימה המעבירה ערך שלם דרך כל אחת מקשתות הרשת.

<u>יעילות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:
בשלב הראשון נעדכן את הזרימה, כך שהיא תתאים לשינוי בקשת e_8. אם הקשת הזאת לא היתה רוויה אז לא נשנה את הזרימה. אם הקשת הזאת היתה רוויה אז נקטין את הזרימה בה ביחידה אחת. נניח שקשת זו מחברת את צומת u לצומת v: נמצא מסלול שיוצא מצומת v אל t ושלאורכו יש זרימה ונקטין את הזרימה בו ביחידה אחת ונמצא מסלול מצומת s ל u שלאורכו מוזרמת זרימה וגם במסלול זה נקטין את הזרימה ב 1.
בשלב השני נעדכן את הזרימה ברשת שמתקבלת לאחר כל השינויים. נבצע איטרציות שבכל אחת מהן ננסה להוסיף יחידת זרימה אחת מ s ל t. יהיו לכל היותר 8 איטרציות כאלה.
הזרימה המתקבלת בשלב הראשון יכולה להיות אופטימלית אם הקטנת הקיבול של קשת e_8 הקטינה את החתך המינימלי. אם היא לא אופטימלית אז היא קטנה רק ביחידה אחת מזרימה אופטימלית. תוספת כוללת של 7 יחידות לא יכולה להגדיל את גודל החתך המינימלי ביותר מ 7 יחידות. לכן לא נזדקק ליותר מ 8 איטרציות בשלב השני. כל עדכון מתבצע על-ידי אלגוריתם BFS.

<u>שאלה 56</u> **(מרצים: פרופ' נוגה אלון, פרופ' יוסי עזר, פרופ' רון שמיר, מתרגלים: רני הוד, אדם שפר)**

נתונים גרף מכוון אציקלי[1] $G=(V,E)$ ופונקצית משקל[2] על הצמתים $\omega : V \to R$, עבור זוג צמתים $u,v \in V$ נסמן $p(u,v)$ את מספר המסלולים השונים מ־ u אל v ב־ G (המסלולים לאו דווקא זרים בקשתות). נגדיר פונקציה נוספת על הצמתים

$$\alpha(u) = \sum_{v \in V \setminus \{u\}} p(u,v) \cdot \omega(v)$$

תארו אלגוריתם אשר מחשב את ערכי α של כל צמתי הגרף ואשר זמן הריצה שלו הינו $O(|V|+|E|)$ (ערכי הפונקציה $p(u,v)$ אינם נתונים).

[1] כלומר, חסר מעגלים מכוונים.

[2] כלומר, ייתכנו גם משקלים חיוביים וגם משקלים שליליים.

<u>אלגוריתם והסבר</u>:

נבצע מיון טופולוגי של צמתי הגרף.
נציב $\alpha(u)=0$ לכל צמתי הגרף.
נעבור על כל צמתי הגרף לפי הסדר ההפוך של המיון הטופולוגי, כך שעוברים על כל צומת לאחר שכבר עברנו על כל הבנים שלו. כשעוברים על צומת נעבור על כל הקשתות היוצאות ממנו. עבור כל בן v של צומת u נציב $\alpha(u)=\alpha(u)+\omega(v)+\alpha(v)$. הערך הסופי יהיה $\alpha(u)$ שמתקבל כאשר מסיימים לטפל בצומת u.
כל המסלולים שיוצאים מצומת עוברים דרך הבנים שלו. כאשר מטפלים בצומת אז כבר ערכי הפונקציה חושבו סופית עבור כל הבנים שלו.

<u>שאלה 57</u> **(מרצים: פרופ' נוגה אלון, ד''ר איריס רוזנבלום, מתרגלים: סבטלנה אולינצקי, רני הוד)**

נתון גרף לא מכוון $G=(V,E)$ המיוצג ע"י רשימות שכנות. תארו אלגוריתם יעיל ככל האפשר שימצא את המספר הקטן ביותר k כך שאפשר לכסות את כל הקשתות של G ב- k מסילות (לאו דווקא פשוטות) כשכל קשת מופיעה במסילה אחת בדיוק, וימצא גם k מסילות כאלה.

<u>יעילות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:

בכל רכיב קשירות נמצא זיווג כלשהו של הצמתים בעלי דרגה אי זוגית (בכל רכיב קשירות יש מספר זוגי של צמתים בעלי דרגה אי זוגית). נוסיף את קשתות הזיווג לאוסף קשתות הגרף. כעת לכל צמתי הגרף יש דרגה זוגית. בכל רכיב קשירות נמצא מעגל אוילר. כעת נסיר מהמעגלים שהתקבלו את כל קשתות הזיווג. הפתרון שנתן יהיה אוסף המסלולים והמעגלים שהתקבלו לאחר הסרת קשתות הזיווג.
אם ברכיב קשירות מסוים יש $2r$ צמתים בעלי דרגה אי זוגית אז אי אפשר לכסות את קשתות הרכיב בפחות מ r מסלולים (כל צומת בעל דרגה אי זוגית צריך להיות קצה של לפחות מסלול אחד).
לאחר ניתוק קשתות הזיווג יתקבלו ברכיב שבו $2r$ צמתים בעלי דרגה אי זוגית, בדיוק r מסלולים.

<u>שאלה 58</u> **(מרצים: פרופ' נוגה אלון, פרופ' יוסי עזר, פרופ' רון שמיר, מתרגלים: רני הוד, אדם שפר)**

<u>סעיף א'</u>

נתונים גרף מכוון $G=(V,E)$ ופונקצית משקל על הקשתות $\omega : E \to R$, בנוסף, ידוע שאין מעגלים שליליים בגרף. תארו אלגוריתם למציאת המסלול הקל ביותר בגרף $p : u \leadsto v$ ביחס למשקל ω .
שימו לב (1) הצמתים u, v אינם נתונים; (2) ייתכן כי $u = v$ והמסלול p ריק; (3) המסלול p לאו דוקא פשוט- ניתן לחזור על צמתים ו/או קשתות.

<u>יעילות</u>:$O(|V||E|)$

<u>אלגוריתם והסבר</u>:

נסתכל על הגרף המתקבל מהגרף המקורי לאחר שמוסיפים לו שני צמתים s, t וגם קשתות: מצומת s יוספו קשתות במשקל 0 לכל צמתי הגרף המקורי ומכל צמתי הגרף המקורי יוספו קשתות במשקל 0 לצומת t. בגרף זה נמצא מסלול קל ביותר מ s ל t. המסלול שאותו נתן כפתרון הוא מסלול זה שממנו מורדות הקשתות שנוגעות ב s וב t.
המסלול מ s ל t חייב להשתמש במסלול שעובר בצמתים האחרים (או שעובר רק בצומת בודד אחר). בכל אחד מהמסלולים האפשריים, המשקלים של החלקים הנוגעים ב s וב t הם בעלי משקל 0. לכן יבחר המסלול הקל ביותר.

<u>סעיף ב'</u>

נתונים גרף מכוון $G=(V,E)$, פונקצית משקל על הקשתות $\omega : E \to R$, וקשת $e \in E$. בנוסף, ידוע שאין מעגלים שליליים בגרף. תארו אלגוריתם יעיל למציאת המסלול הקל ביותר בגרף מבין המסלולים העוברים דרך הקשת e . המסלול עשוי לבקר יותר מפעם אחת בצמתים ו/או קשתות.

<u>יעילות</u>: $O(|V||E|)$

<u>אלגוריתם והסבר</u>:

נניח שהקשת e היא מהצומת u לצומת v .
נמצא את המסלולים הקלים ביותר מהצומת v לכל צמתי הגרף ונמצא את המסלולים הקלים ביותר מכל צמתי הגרף לצומת u. הפתרון שנתן יורכב ממסלול שהוא קל ביותר מבין המסלולים המסתיימים ב u, מהקשת e וממסלול שהוא קל ביותר מבין המסלולים היוצאים מ v . מציאת משקלי המסלולים שמסתיימים ב u תתבסס על מציאת המסלולים האופטימלים מ u בגרף שכיווני קשתותיו הם הפוכים לכיווני קשתות הגרף המקורי (מבלי שינוי במשקלי הקשתות). לכל מסלול יש חלק (אולי ריק) שמסתיים ב u וחלק (אולי ריק) שמתחיל ב v. לגבי כל אחד מחלקים אלה מצאנו מסלול קל ביותר אפשרי.

<u>שאלה 59</u> (מרצים: פרופ' עמוס פיאט , פרופ' מיכה שריר , מתרגלים: שי ורדי , אדם שפר)

<u>סעיף א'</u>

הוכיחו או הפריכו: קיימת רשת זרימה $G=(V,E)$ אשר מכילה 7 קשתות עם קיבול 1 ועוד 30 קשתות עם קיבול 10, והזרימה המקסימלית בה הינה בגודל 39 (פרט ל 37 הקשתות הנתונות, אין קשתות נוספות ברשת).

<u>פתרון</u>:

לא קיימת.

זרימה מכסימלית שווה לגודל חתך מינימלי. בחתך זה יכולות להיות רק קשתות שקיבולן 1 או 10. אין סכום של כפולה של 7 וכפולה של 10 ששווה ל 39.

<u>סעיף ב'</u>

הוכיחו או הפריכו: קיימת רשת זרימה $G=(V,E)$ אשר כל הקיבולים בה הם 1 או $\sqrt{2}$, קיים ברשת מסלול בין המקור לבור שכל הקשתות שבו בעלות קיבול $\sqrt{2}$, והזרימה המקסימלית הינה בגודל 39.

<u>פתרון</u>:

לא קיימת.

לגבי כל חתך ובפרט חתך מינימלי, יש שני צמתים עוקבים במסלול שנמצאים בצדדים שונים של החתך (במסלול הנתון, המקור הוא בצד אחד והבור הוא בצד שני. לכן חייבת להיות קשת אחת שעוברת מהצד הראשון לצד השני). לכן בין שני צידי החתך יש לפחות קשת אחת שקיבולה $\sqrt{2}$. לכן גודל החתך אינו שלם.

<u>סעיף ג'</u>

הוכיחו או הפריכו: קיימת רשת זרימה $G=(V,E)$ אשר כל הקיבולים בה הם 1 או $\sqrt{2}$, קיים ברשת חתך (לאו דווקא מינימלי) מהמקור אל הבור שכל הקשתות שלו בעלות קיבול $\sqrt{2}$ והזרימה המקסימלית היא 39.

<u>פתרון</u>:

כן קיימת.

נתאר דוגמא של גרף שבו $2\cdot 39=78$ קשתות. נניח שמצומת s יש 39 קשתות בעלות קיבול $\sqrt{2}$ ל 39 צמתים שונים. מכל אחת מ 39 צמתים אלה יש קשת בעלת קיבול 1 ל t. הזרימה המכסימלית היא בדיוק 39.

<u>שאלה 60</u> (מרצים: פרופ' עמוס פיאט , פרופ' מיכה שריר , מתרגלים: שי ורדי , אדם שפר)
<u>סעיף א'</u>
נתונים גרף מכוון $G=(V,E)$, פונקצית משקל על הקשתות $w:E\to R$, וזוג קודקודים $s,t\in V$.
ידוע שאין ב G מעגל שלילי. בנוסף, 10 מקשתות הגרף צבועות באדום. תארו אלגוריתם יעיל למציאת מסלול קל ביותר מ s אל t, מבין המסלולים אשר מכילים לפחות חמש קשתות אדומות. מעבר דרך אותה קשת אדומה x פעמים נספר כ x קשתות אדומות במסלול, ולא כאחת.

<u>יעילות</u>: של חישוב מרחק מינימלי ללא המגבלה שצריך לפחות 5 קשתות אדומות

<u>אלגוריתם והסבר</u>:
נשתמש בגרף של שכבות. בכל שכבה יהיה עותק של הגרף. בין כל שתי שכבות עוקבות יופיעו רק הקשתות האדומות. נחשב מרחק מינימלי בין הצומת s שבשכבה הראשונה לצומת t שבשכבה השישית. כך חייבים להשתמש 5 פעמים בקשתות אדומות. אפשר לעשות שימוש נוסף בקשתות אדומות בתוך שכבה.

<u>סעיף ב'</u>
נתונים גרף מכוון $G=(V,E)$, פונקצית משקל על הקשתות $w:E\to R$, וזוג קודקודים $s,t\in V$.
ידוע שאין ב G מעגל שלילי. בנוסף, 10 מקשתות הגרף צבועות באדום. תארו אלגוריתם יעיל למציאת מסלול קל ביותר מ s אל t, מבין המסלולים אשר מכילים לפחות חמש קשתות אדומות. מעבר דרך אותה קשת אדומה x פעמים נספר **כקשת אדומה אחת**, ולא כ x קשתות אדומות במסלול.

<u>יעילות</u>: של חישוב מרחק מינימלי ללא המגבלה שצריך לפחות 5 קשתות אדומות

<u>אלגוריתם והסבר</u>:
נחשב כמה פתרונות. הפתרון שנתן יהיה הטוב מבין הפתרונות המחושבים. בכל פתרון נבחר קבוצה סדורה של 5 קשתות אדומות מתוך ה 10 קשתות אדומות. נחשב את המרחק מ s לכניסה של הראשון שבינהם, מכל יציאה של קשת את המרחק לכניסה של הקשת הבאה ומהיציאה של האחרון לכניסה של הבא. נסכם את ששת האורכים האלה כולל אורכי חמשת הקשתות שבחרנו.
כך נקבל מסלול שעובר בלפחות 5 קשתות אלה. כך ננסה את כל החמשיות הסדורות מתוך עשרת הקשתות האדומות.

<u>שאלה 61</u> (מרצים: פרופ' עמוס פיאט , פרופ' מיכה שריר , מתרגלים: שי ורדי , אדם שפר)

נתונים בניין עם n קומות ו m אבטיחים. ידוע שקיים מספר x כלשהו, כך שאם נזרוק אבטיח מקומה x (או מקומה נמוכה יותר ממנה) האבטיח ישרוד את הנפילה, אך אם נזרוק את האבטיח מקומה גבוהה מ x האבטיח יתנפץ. באבטיח ששרד ניתן להשתמש לזריקות נוספות, אך לא באבטיח שהתנפץ. תארו אלגוריתם יעיל שמוצא, עבור n,m נתונים, את מספר זריקות האבטיחים המינימלי w שנזדקק להן על מנת שבטוח נגלה את הערך x. הניתוח אמור להיות ביחס למקרה הגרוע ביותר. כלומר, עבור כל תוצאה אפשרית שתתקבל מהזריקות שביצענו, נזדקק לכל היותר ל w זריקות כדי לקבוע מהו x. סדרת הקומות שבהן נשתמש אינה צריכה להיות קבועה מראש- כל קומה יכולה להבחר לאחר שידועות התוצאות של הזריקות הקודמות. רמז: הגדירו את $c[i,j]$ להיות מספר הזריקות המינימלי שנזדקק לו כאשר בבניין יש i קומות ויש לנו j אבטיחים.

<u>יעילות</u>: $O(n^2m)$ (אם m גדול מספיק ביחס ל n אז היעילות היא $O(1)$)

<u>אלגוריתם והסבר</u>:

נשתמש בתכנות דינמי.

עבור כל $1 \le i \le n$, $0 \le j \le m$ נגדיר $c[i,j]$ המחושבים לפי

עבור $2 \le i \le n$, $1 \le j \le m$: $c[i,j] = \min_k\{\max\{c[k-1,j-1]+1, c[n-k,j]+1\}\}$

כאשר $c[1,j]=0$ לכל j ו $c[i,0]=n+1$ לכל $i>1$

ומבוקש $c[n,m]$.

כאשר ידוע שהגובה המכסימלי האפשרי להשרדות הוא אחד מבין i אפשרויות אז אם בוחרים לבדיקה את השלב ה- k מביניהם אז אם האבטיח לא ישרוד אז ישארו $k-1$ אפשרויות לאחר שבזבזנו אבטיח וגם בדיקה. אם האבטיח ישרוד אז ישארו $n-k$ אפשרויות מבלי שבזבזנו אבטיח, אבל כן בזבזנו בדיקה. אסור להישאר בלי אבטיחים כאשר צריך עדיין לבצע בדיקות. רשמנו שבמצב זה יש צורך ב $n+1$ בדיקות שזה יותר ממה שנחוץ לעשות בדיקות עם אבטיח בודד (תמיד אפשר עם אבטיח בודד לטפס בכל שלב קומה אחת עד שהוא יתנפץ).

אם יש לנו לפחות $\log_2(n)+1$ אבטיחים אז הבדיקה היעילה ביותר היא על-ידי שבכל שלב נטיל אבטיח באמצע הקטע הנותר. כך החסם על מספר ההטלות ידוע מראש.

<u>שאלה 62</u> (מרצים: פרופ' נוגה אלון, רני הוד, אדם שפר, מתרגלים: רני הוד, שי ורדי)

נתונות שתי קבוצות של מספרים טבעיים $X,Y \subseteq \{1,2,\cdots,100n\}$ כך ש $|X|=3n$ ו $|Y|=7n$.

תארו אלגוריתם יעיל שיבדוק האם יש תת-קבוצות $A \subseteq X, B \subseteq Y$ לא ריקות כך ש- $|A|=|B|$ וכן סכום איברי A שווה לסכום איברי B .

<u>יעילות</u>: $O(n^4)$

<u>אלגוריתם והסבר</u>:

הכוונה היא למצוא בעזרת תכנות דינאמי את כל הסכומים שניתן ליצור מכל אחד משני האוספים עבור כל עוצמת קבוצה, ואחר-כך לעבור על כל אחד מהסכומים האפשריים בצירוף עם העוצמות האפשריות ולבדוק אם הוא נוצר על-ידי כל אחד מהאוספים עם אותה עוצמה. יש סדר גודל של n אפשרויות לגבי העוצמות שיוצרים סכום מסוים. יש סדר גודל של n^2 של סכומים שיכולים להיוצר. לכן יש בסך הכל סדר גודל של n^3 אפשרויות.

עבור כל אחד משני האוספים, בתחילה הטבלה שמימדיה הם בסדרי גודל של n על n^2 אפשרויות מכילה ערכי שקר ורק האיבר שבה של 0 מספרים וסכום 0 הוא בעל ערך אמת. בשלב השני נעבור על פני המספרים השונים. עבור כל מספר נראה אילו צירופים שעדיין לא היו קיימים, יכולים להיוצר בעזרתו. אם גודל המספר הוא d, אז אם עבור טבעי מסוים k, היה ניתן ליצור סכום A בעזרת k מספרים, אז כעת ניתן ליצור גם את הסכום $A+d$ בעזרת $k+1$ מספרים. האיברים המתאימים יעודכנו לערכי אמת.

נציג וריאציה של האלגוריתם:

נבצע טרנספורמציה על כל אחד מהמספרים שבשני האוספים. לכל מספר טבעי נוסיף $1000n^2$ (1000 הוא בחירה שרירותית). כך לא יהיו שני אוספים שווי משקל בעלי עוצמות שונות. בשלב השני נחשב באמצעות תכנות דינאמי את כל הסכומים החיובים שניתן ליצור בכל אחד מהאוספים (לגבי כל אוסף נפעיל תכנות דינאמי נפרד שבו כאשר נעבור על מספר מסוים, נראה אילו מספרים ניתן ליצור מהסכומים שהיו עד כה ובתוספת שלו). כל הסכומים שיוצרו הם קטנים מלמשל 10^6n^3. בשלב השלישי נעבור על כל הטבעיים שבין 1 ל 10^6n^3 ונראה אם יש ביניהם טבעי שנוצר גם בקבוצה הראשונה וגם בקבוצה השניה.

בשלב השני יש בכל קבוצה n איטרציות שבכל אחת מהן עוברים על סדר גודל של n^3 סכומים שכבר נוצרו.

<u>שאלה 63</u> (מרצים: פרופ' נוגה אלון, רני הוד, אדם שפר, מתרגלים: רני הוד, שי ורדי)

נתונה רשת זרימה $G=(V,E)$ מ- s ל- t עם קיבולים שלמים $c:E\to Z_{\ge 0}$. תארו אלגוריתם יעיל שימצא האם יש ברשת חתך מקיבול מינימלי עם לכל היותר 100 קשתות.

<u>יעילות</u>: של אלגוריתם לחישוב זרימה מכסימלית

<u>אלגוריתם והסבר</u>:

נריץ אלגוריתם לזרימה מכסימלית מהמקור ליעד בגרף הנתון. נריץ גם אלגוריתם לזרימה מכסימלית על הגרף שבו קיבול כל קשת שווה לקיבולה המקורי ועוד הקבוע $a=\frac{1}{2|e|}$. מספר הקשתות בחתך מינימלי שבו מספר מינימלי של קשתות נתון על-ידי הפרש ערכי שני הפתרונות חלקי a. כך נוכל לדעת אם הוא לא גדול מ 100.

הקיבול בכל חתך מינימלי יגדל ב a כפול מספר הקשתות שבו. לגבי חתך שאינו מינימלי ברשת המקורית, גודלו היה לפני השינוי גדול בלפחות 1 מאשר חתך מינימלי . בכל חתך אין יותר מ $|e|$ קשתות. לכן חתך מינימלי ישאר בעל קיבול נמוך יותר מאשר כל חתך שאינו מינימלי. כך שום חתך שלא היה מינימלי לא יהפוך לחתך מינימלי.

<u>שאלה 64</u> **(מרצים: פרופ' נוגה אלון, פרופ' רון שמיר, מתרגלים: רני הוד, שי ורדי)**
נתון מספר טבעי k, גרף מכוון $G=(V,E)$ עם פונקציית משקל חיובית $\omega: E \to R_+$ ונתונים זוג צמתים $x, y \in V$.
תארו אלגוריתם יעיל ככל הניתן שימצא את קבוצת כל הקשתות שמשתתפות ב-k מק"בים מ-x ל-y;
במילים אחרות, הפלט הוא קבוצת הקשתות $e \in E$ כך שמספר המסלולים השונים מ-x ל-y ממשקל $\delta(x,y)$ שעוברים דרך e הוא לפחות k, כאשר $\delta(x,y)$ הוא משקל המסלול הקל ביותר מ-x ל-y (ביחס ל-ω).

<u>יעילות</u>: של אלגוריתם דיקסטרה

<u>אלגוריתם והסבר</u>:
באמצעות אלגוריתם דיקסטרה נחשב את המרחקים לכל צמתי הגרף מצומת x.
באמצעות אלגוריתם דיקסטרה נחשב את המרחקים מכל צמתי הגרף לצומת y (נעשה זאת על-ידי הרצת האלגוריתם מהצומת y על הגרף שכיווני קשתותיו הפוכים למקוריים).
נמיין את הצמתים לפי מרחקם מהצומת x.
בשלב הבא (שהוא בדיקה אילו קשתות נמצאות על עץ המרחקים הקצרים ביותר): עבור כל קשת (u,v) נבדוק אם היא נמצאת על מסלול קצר ביותר מ x ל y (נבדוק אם הסכום של המרחק מ x ל u בתוספת משקל הקשת ובתוספת המרחק מ v ל y נותנים את המרחק מ x ל y). מעתה נתייחס רק לקשתות שנמצאות על מסלול קצר ביותר. את יתר הקשתות נמחק.
בשלב הבא עבור כל קשת (u,v) נחשב את מספר המסלולים הקצרים ביותר מ x ל u ואת מספר המסלולים הקצרים ביותר מ v ל y. המכפלה של המספרים האלה היא מספר המסלולים הקצרים ביותר שעוברים דרך הקשת הזאת. נבדוק עבור אילו קשתות מכפלה זו גדולה מ k.
חישוב מספר המסלולים הקצרים ביותר מ x ל u: בתחילה המונה עבור הצומת x הוא 1 ועבור יתר הצמתים הוא אפס. נעבור על צמתי הגרף לפי הסדר שהם מוינו. עבור כל צומת, מספר המסלולים יהיה שווה לסכום מספרי המסלולים של כל הצמתים שמהם יש קשת ל u. עבור כל צומת שמוביל ל u נוסיף למונה של u את מספר המסלולים אל צומת זה.
חישוב מספר המסלולים הקצרים ביותר מ v ל y יעשה בדרך דומה. הפעם בתחילה רק לגבי הצומת y המונה יהיה שווה ל 1. נעבור על הצמתים בסדר יורד. מספר המסלולים הקצרים ביותר מ v ל y שווה לסכום מספרי המסלולים הקצרים ביותר שעוברים דרך צמתים שיש מ v קשת אליהם.

<u>שאלה 65</u> **(מרצים: פרופ' נוגה אלון, פרופ' רון שמיר, מתרגלים: רני הוד, שי ורדי)**
נתון גרף לא מכוון $G=(V,E)$ וידוע ש G <u>אינו</u> דו-צדדי. להלן אלגוריתם עבור בעיית הזיווג המקסימלי: נתבונן בגרף החילה (incidence) H של G. זהו גרף דו-צדדי שצמתיו הם V בצד האחד ו-E בצד השני. עבור $e \in E, v \in V$, v ו-e הם שכנים ב-H אם"ם v הוא קצה של e. כעת נהפוך את H לרשת זרימה באופן הבא:

- נוסיף צומת מקור s שיחובר לכל $v \in V$ ע"י קשת מקיבול 1;
- נכוון את קשתות H מ-V אל E וניתן קיבול 1 לכולן;
- נוסיף צומת בור t שכל $e \in E$ תחובר אליו ע"י קשת מקיבול 2.

כעת נחשב זרימה מקסימלית f^* מ-s ל-t ברשת.
נסמן את גודל הזיווג המקסימלי בגרף G ב-m. הוכיחו או הפריכו כל אחת מהטענות הבאות:

(א) $m \ge \frac{1}{2}|f^*|$ (ב) $m \le \frac{1}{2}|f^*|$

<u>הוכחה/דוגמא נגדית</u>:

<u>א</u>. נפריך את הטענה באמצעות דוגמא נגדית.
הדוגמא היא של גרף שהוא משולש. גודל זיווג מקסימלי הוא 1.
גודל הזרימה הוא 3.

<u>ב</u>. הטענה נכונה.
גודל זיווג מקסימלי הוא אף פעם לא יותר ממחצית מספר צמתי הגרף שנוגעים בקשת אחת לפחות.
עבור כל צומת שנוגע בלפחות קשת אחת, ניתן להזרים דרכו יחידה אחת, בין אם ניתן או לא ניתן לזווגו לצומת אחר. לכן גודל הזרימה f^* שווה למספר הצמתים שנוגעים בקשת אחת לפחות.

<u>שאלה 66</u> (מרצה: פרופ' עמוס פיאט, מתרגלים: שי ורדי, אילן כהן)
נתון גרף מכוון $G=(V,E)$ עם פונקצית משקל $\omega: E \to R$, ידוע שאין מעגלים שלילים בגרף.
בהינתן תת-קבוצה של הקודקודים $U \supseteq V$, תארו אלגוריתם יעיל שמחשב לכל זוג צמתים $v,w \in V$ את משקל המסלול הקל ביותר בין v ל w שעובר דרך **לפחות** אחד מקודקודי U, וחשבו את סיבוכיות זמן הריצה של האלגוריתם. הניחו $|U| = \Omega(|V|)$.

<u>אלגוריתם והסבר</u>:
יהיו שלושה עותקים של צמתי הגרף. עבור כל זוג צמתים $v_1, v_2 \in V$ שיש קשת מ v_1 ל v_2 בגרף המקורי, יהיו קשתות בעלות משקל זהה למשקל הקשת המקורית, בין הנציג של v_1 שבעותק הראשון לבין הנציג של v_2 שבעותק הראשון ובין הנציג של v_1 שבעותק השלישי לבין הנציג של v_2 שבעותק השלישי. עבור כל זוג צמתים $v_1 \in V$, $u_2 \in U$ שיש קשת מ v_1 ל u_2 בגרף המקורי, תהיה קשת בעלת משקל זהה מהנציג של v_1 שבעותק הראשון לנציג של u_2 שבעותק השני. עבור כל זוג צמתים $u_1 \in U$, $v_2 \in V$ שיש קשת מ u_1 ל v_2 בגרף המקורי, תהיה קשת בעלת משקל זהה בין הנציג של u_1 שבעותק השני לבין הנציג של v_2 שבעותק השלישי.
נריץ אלגוריתם לחישוב המרחקים בין כל זוגות הצמתי שבגרף שבנינו.
עבור כל זוג צמתים $v,w \in V$ המרחק המבוקש בין v ל w יהיה שווה למרחק שבין הנציג של v שבעותק הראשון לבין הנציג של w שבעותק השלישי.
לכל מסלול שבין שני צמתים יש ייצוג על-ידי מסלול מהעותק הראשון לעותק השלישי. בתוך העותקים הראשון וגם השלישי מופיעות כל הקשתות של הגרף המקורי. באיזשהו מקום צריך לעבור דרך העותק השני. אפשר לעבור בנוסף גם בנציגים נוספים של צמתים מ U במעברים שבתוך העותק הראשון ובתוך העותק השלישי.
כל מסלול שקיים בגרף שבנינו מייצג גם מסלול בגרף המקורי.
לא שינינו את סדרי הגודל של מספרי הצמתים והקשתות ולכן הסיבוכיות שווה לסיבוכיות של אלגוריתם למציאת כל המרחקים בגרף המקורי.

<u>שאלה 67</u> (מרצה: פרופ' עמוס פיאט, מתרגלים: שי ורדי, אילן כהן)
נתון גרף $G=(V,E)$ עם פונקצית משקל $\omega: E \to R$. תארו אלגוריתם יעיל שמחשב לכל צומת $v \in V$, האם v נמצא במעגל שלילי כלשהוא. (לא בהכרח מעגל פשוט).

<u>אלגוריתם והסבר</u>:
נחשב את רכיבי הקשירות החזקה בגרף. בכל רכיב קשירות חזקה, נבדוק אם יש מעגל שלילי. נעשה זאת על-ידי חישוב המרחקים מצומת שרירותי ברכיב אל יתר צמתי הרכיב באמצעות אלגוריתם בלמן פורד.
אם יש מעגל שלילי ברכיב קשירות חזקה מסוים, אז ניתן להגיע למעגל השלילי ולעבור עליו מספר פעמים כנדרש ומשם להמשיך בחזרה. כך כל צומת ברכיב זה נמצא על מעגל שלילי.
אם ברכיב יש מעגל שלילי, אז יהיה שיפור במרחק לאיזשהו צומת באיטרציה האחרונה, כאשר מספר האיטרציות שווה למספר הצמתים שברכיב. שיפור כזה מצביע על קיומו של מעגל שלילי ברכיב.
במעגל לא משתתפים צמתים מרכיבי קשירות חזקה שונים.
<u>הערה</u>: מכיון שכל צומת וכל קשת שייכים ללכל היותר לרכיב קשירות חזקה אחד ומכיון שבכל רכיב קשירות חזקה חישבנו מרחקים רק מצומת אחד, אז סיבוכיות האלגוריתם היא הסיבוכיות של אלגוריתם בלמן פורד ולא של אלגוריתם למציאת מרחקים בין כל זוג צמתים.

<u>שאלה 68</u> (מרצים: פרופ' עמוס פיאט, פרופ' מיכה שריר, מתרגלים: שי ורדי, אילן כהן)
נתון גרף פשוט ולא מכוון $G=(V,E)$ עם פונקציית משקל של הקשתות $\omega: E \to \{-1,0,1\}$, וצומת s . תארו אלגוריתם יעיל ככל האפשר אשר מוצא, מבין כל העצים שמהווים עץ מסלולים קצרים מ- s <u>ביחס למספר הקשתות</u>, את העץ עם סכום המשקולות המינימאלי.

<u>סיבוכיות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:
בשלב הראשון נריץ אלגוריתם BFS מהצומת s למציאת המרחקים ביחס לספירת קשתות מהצומת s .
כך הצמתים יחולקו לשכבות לפי המרחק שלהם מ s בספירת קשתות. נשמור את כל הקשתות שמחברות צמתים בשכבות עוקבות.
בשלב השני נעבור על צמתי השכבות השונות בסדר עולה. ההורה לכל צומת, יהיה צומת מהשכבה הקודמת, שהקשת ביניהם היא בעלת משקל מינימלי.
המסלול חייב להגיע ישירות מהשכבה הקודמת, כי אחרת, הוא לא קצר ביותר בספירת קשתות.

<u>שאלה 69</u> (מרצים: פרופ' נוגה אלון, פרופ' עמוס פיאט, פרופ' רון שמיר, מתרגלים: שי ורדי, אילן כהן)

יהי $G=(V,E)$ גרף פשוט ומכוון עם פונקצית משקל על הקשתות $\omega : E \to R^{+}$. כמו כן, כל קודקוד צבוע בצבע כלשהו. נתונים זוג קודקודים $s,t \in V$, לא בהכרח שונים, הצבועים בכחול. תארו אלגוריתם יעיל ככל האפשר המוצא את המסלול הקל ביותר מ- s ל- t שעובר דרך לפחות קודקוד אדום אחד, ולא מכיל רצף של יותר מ-4 קודקודים אדומים.

<u>סיבוכיות</u>: של אלגוריתם דיקסטרה

<u>אלגוריתם והסבר</u>:
נבנה גרף בעל מספר שכבות. בשכבות 1 ו 6 יופיעו רק הקודקודים שאינם אדומים. בשכבות 2,3,4,5 יופיעו רק הקודקודים האדומים. כל קשת מהגרף המקורי שיוצאת מקודקוד שאינו אדום ומגיעה לקודקוד כלשהו, תופיע בין הנציגים של אותם קודקודים שבאותה שכבה ובין הנציגים שלהם משכבה 1 לשכבה 2 ובין הנציגים שלהם משכבה 6 לשכבה 1, כל קשת שבין קודקוד אדום לקודקוד כלשהו, תופיע בין הנציגים של אותם קודקודים שבשכבות 2 ל 3, 2 ל 6 , 3 ל 4, 3 ל 6, 4 ל 5, 4 ל 6, 5 ל 6.
נמצא מרחק מינימלי בין הנציג של s שבשכבה 1 לנציג של t שבשכבה 6.
עבור כל מסלול, יש ייצוג לכל חלק שלו שעובר רק בין קודקודים שאינם אדומים. ניתן לשבץ גם קודקודים אדומים, אך לא יותר מרצף של ארבעה אדומים. כדי להגיע לשכבה 6, חייבים לעבור לפחות פעם אחת בשכבה 2.
לא שינינו את סדר הגודל של מספרי הקודקודים והקשתות.

<u>שאלה 70</u> (מרצים: פרופ' נוגה אלון, פרופ' עמוס פיאט, פרופ' רון שמיר, מתרגלים: שי ורדי, אילן כהן)

נתון גרף פשוט, קשיר ולא מכוון $G=(V,E)$ עם פונקציית משקל על הקשתות $\omega : E \to R$. נתון כי $|V|=n, |E|=n+10$. תארו אלגוריתם יעיל ככל האפשר המוצא עץ פורש מינימאלי של G.

<u>סיבוכיות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:
בכל אחד מ 11 השלבים, נסיר מהגרף קשת שהיא בעלת משקל מכסימלי במעגל כלשהו.
בכל שלב נריץ BFS מצומת שרירותי. עבור כל צומת שאליו נגיע, נשמור את האב שלו בעץ ה BFS.
באיזשהו שלב נגיע לצומת משני אבות. כאשר זה יקרה, יהיו לנו שתי סדרות של אבות ומשקולות של קשתות מאב לבן. יש אב בעל מספור מכסימלי שמשותף לשתי סדרות אלה. מאוסף הקשתות שהן חלק מתת העץ שמתחיל בצומת זה ומוביל לצומת שאליו הגענו בשתי דרכים, נסיר את הקשת בעלת משקל מכסימלי. קשת כזאת נמצאת על מעגל. לפי הוכחת הנכונות של אלגוריתם קרוסקל, קיים עץ פורש מינימאלי שלא מכיל קשת שהיא בעל משקל מכסימלי במעגל.

<u>שאלה 71</u> (מרצים: פרופ' נוגה אלון, פרופ' עמוס פיאט, פרופ' רון שמיר, מתרגלים: שי ורדי, אילן כהן)

יהי $G=(V,E)$ גרף פשוט ומכוון שבו כל קודקוד צבוע בצבע כלשהו. מסלול k-מתחלף הוא מסלול אשר צבעי הקודקודים מתחלפים לאורכו k פעמים. לדוגמא, מסלול שבו צבעי הקודקודים הם **אדום-כחול-אדום-אדום-ירוק-ירוק-כחול** הוא 4-מתחלף. תארו אלגוריתם יעיל ככל האפשר המוצא את ה-k המקסימאלי עבורו קיים מסלול ב-G או מחזיר ∞ אם לא קיים k כזה.

<u>סיבוכיות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:

<u>הערה</u>: מכך שקיימת אפשרות שלא קיים k מכסימלי, אפשר להסיק שהכוונה היא שמסלול יכול לחזור על קשתות. אחרת הוא מוגבל באורכו.

נקרא לקשתות שבין שני צמתים שוני צבע- קשתות מיוחדות. נמצא את רכיבי הקשירות החזקה בגרף. אם יש רכיב קשירות חזקה שבו קשת מיוחדת, אז הפתרון הוא ∞. אחרת, נבנה גרף עזר לפי הגרף המקורי. נתאר את תהליך הבניה: תחילה כל גשר יפוצל למסלול באורך 2 שמחבר בין שני רכיבי הקשירות שהגשר המקורי חיבר. לגבי הגרף המתקבל, נחשב את גרף העל. צמתי גרף העל יקבלו משקלים של 0 ו 1. צמתים שמייצגים רכיב קשירות חזקה יקבלו משקל 0 (כאמור לא היתה בהם קשת מיוחדת). צומת שמייצג קשת שהיתה גשר יקבל משקל 1 אם"ם הקשת המקורית היתה מיוחדת. כעת נחשב על גרף העל שהתקבל, מסלול בעל משקל מכסימלי. עבור צומת i שהוא עלה יתקיים $F(i)=w(i)$ ועבור צמתים אחרים יתקיים ש $F(i)$ שווה למכסימום על פני j שלהם יש קשת מ i של $F(j)+w(i)$, כאשר w היא פונקצית המשקל שמקבלת ערכים של 0 ו 1.

<u>שאלה 72</u> (מרצים: פרופ' נוגה אלון, ד"ר רני הוד, מתרגלים: אופיר פרידלר, אלון עדן)

נתון גרף מכוון $G=(V,E)$ שבו לכל קשת צבע אדום או כחול, ונתונים זוג צמתים שונים $s,t\in V$. תארו אלגוריתם יעיל ככל האפשר שימצא מסלול מכוון מ-s ל-t שמספר שינויי הצבע לאורכו הוא מינימלי, או שיודיע שלא קיים מסלול כזה.

<u>יעילות</u>: $O(|V|+|E|)$

<u>אלגוריתם והסבר</u>:

נבנה גרף עזר. את כל הצמתים חוץ מ s ומ t נפצל לשניים. לנציג הראשון יכנסו ויצאו כל הקשתות האדומות שנוגעות בצומת המקורי. לנציג השני יצאו ויכנסו כל הקשתות הכחולות של הצומת המקורי. בין כל שני נציגים של אותו צומת יהיו בשני הכיוונים קשתות. הצמתים s ו t יחוברו לנציגים ומהנציגים המקוריים של שכניהם המקוריים בגרף. נבצע סריקה של צמתי הגרף החל מהצומת s. ברגע שנגיע לצומת שמייצג צומת מסוים, נכניס לאותה שכבה את כל הצמתים שאליהם יוצאת ממנו קשת בצבע זה ושעדיין לא הגענו אליהם. לאחר שנסיים להוסיף את כל הצמתים שבשכבה זו, נתייחס לצומת שהוא הנציג האחר של הצומת המקורית שלו, אם הוא עדיין לא מופיע בשכבה ששווה לכל היותר לשכבה שלו אז נכניס אותו לשכבה העוקבת. לגבי כל צומת שאנו מעדכנים את שכבתו, נסמן את ההורה שלו, כך שלאחר שנגיע לצומת t, נוכל לשחזר מסלול אופטימלי. בתהליך זה הופעה בשכבה מסוימת מייצגת מספר מסוים של החלפות של צבעים. בהינתן שעד שכבה מסוימת כל המרחקים מעודכנים, אז לצמתים שנמצאים בשכבה הבאה, צריך להגיע על-ידי החלפת צבע.

<u>שאלה 73</u> **(מרצים: פרופ' נוגה אלון, ד''ר רני הוד, מתרגלים: אופיר פרידלר, אלון עדן)**
נתון גרף מכוון $G=(V,E)$ עם פונקציית משקל על הקשתות $\omega: E \rightarrow R$, $r \in V$. בנוסף נניח שאין מעגלים שליליים. בנוסף ידוע שאין בגרף מסלול פשוט שמספר קשתותיו גדול מ- $\sqrt{|V|}$. תארו אלגוריתם יעיל ככל האפשר שימצא עץ מסילות קלות ביותר מהצומת r.

<u>יעילות</u>: $O\left(|E|\sqrt{|V|}\right)$

<u>אלגוריתם והסבר</u>:
נבנה גרף של $\left\lfloor\sqrt{|V|}+1\right\rfloor$ שכבות. בשכבה הראשונה יופיע רק הצומת r ובכל אחת מהשכבות האחרות יופיעו נציגים של כל אחד מהצמתים האחרים. מכל נציג של כל צומת יצאו קשתות לצמתים שנמצאים בשכבה הבאה אל הנציגים של צמתים שיש לצומת המקורי קשת אליהם. לקשתות אלה יהיה משקל ששווה למשקלן בגרף המקורי. כמו כן יהיו קשתות במשקל אפס מכל נציג אל הנציג של אותו צומת מקורי שנמצא בשכבה הבאה. לאחר שחישבנו את המרחקים לכל הצמתים בשכבה מסוימת, נחשב את המרחקים לצמתי השכבה הבאה. לגבי כל נציג, המרחק אליו יהיה שווה למינימום של המרחקים דרך צמתים מהשכבה הקודמת, כאשר המרחק שווה לסכום של המרחק לצומת מהשכבה הקודמת בתוספת משקל הקשת שביניהם. כדי שנוכל לשחזר מסלולים, אז עם כל עידכון, נסמן לגבי כל צומת את ההורה שממנו הגענו אליו. המסלול לצומת מסוים יהיה המסלול שמגיע לנציגו שבשכבה האחרונה. הקשתות שבין נציגים של אותו צומת מקורי ייצגו השארות באותה צומת. הוכחת האופטימליות של המרחקים המתקבלים, מתקבלת באינדוקציה על פני השכבות השונות, כאשר המרחק לנציג מסוים הוא המרחק האופטימלי המתקבל עם הגבלה במספר הקשתות.

<u>שאלה 74</u> **(מרצים: פרופ' נוגה אלון, ד''ר רני הוד, מתרגלים: אופיר פרידלר, אלון עדן)**
נתון גרף לא מכוון $G=(V,E)$ ופונקציית משקל $\omega: E \rightarrow \{1,2,\cdots,|V|\}$ המתאימה לכל קשת משקל שלם. תארו אלגוריתם יעיל ככל האפשר המוצא את אוסף כל הקשתות e של G עבורן יש מעגל C ב- G שמכיל את e כך שלכל קשת $e' \in C$, $e' \neq e$. מתקיים ש $\omega(e) > \omega(e')$.

<u>יעילות</u>: של "יוניון פינד"

<u>אלגוריתם והסבר</u>:
נריץ את האלגוריתם של קרוסקל לחישוב עץ פורש מינימלי עם מספר שינויים.
את מיון הקשתות על פי משקלן נבצע בזמן לינארי על-ידי הכנסת כל קשת לתור של קשתות בעלות משקל זהה לשלה.
כאשר בשלב השני של האלגוריתם נגיע למשקל מסוים, נבדוק לגבי כל קשת בעלת משקל כזה, אם היא יכולה להכנס לפתרון כאשר היא בעלת העדיפות הראשונה מבין הקשתות שבמשקל זה. בשלב זה עדיין לא נוסיף קשתות לפתרון. אחרי שנסיים את הבדיקות לגבי משקל מסוים, נתחיל עם תהליך בדיקות והכנסות רגילות של קשתות ממשקל זה. כל קשת שיכלה להכנס לפתרון אינה נמצאת על מעגל שכל קשתותיו קלות יותר. השרירותיות שבבחירת הקשתות שלמעשה נכנסות לפתרון אינה משפיעה על הבדיקות שנעשות בהמשך, כי קשת שיכלה להבחר ולא נבחרה, מחברת בין שני צמתים שכבר נמצאים באותו רכיב קשירות.

<u>שאלה 75</u> **(מרצים: פרופ' נוגה אלון, ד"ר רני הוד, מתרגלים: אופיר פרידלר, אלון עדן)**
נתון גרף מכוון $G=(V,E)$ ופונקצית משקל $\omega : E \to Z$. תארו אלגוריתם יעיל ככל האפשר שמוצא את אוסף כל הצמתים v שיש מהם מסילה מכוונת אל מעגל שלילי ב- G.

<u>יעילות</u>: $O(|V||E|)$

<u>אלגוריתם והסבר</u>:
נחשב רכיבי קשירות חזקה בגרף.
לגבי כל רכיב קשירות חזקה, נוסיף לגרף צומת שממנו יצאו קשתות בעלות משקל אפס לכל צמתי הרכיב. נחשב באמצעות אלגוריתם בלמן פורד מרחקים מהצמתים שהוספנו לכל צמתי הרכיב שאליהם הם מחוברים. מכל צמתי רכיב שלגביו התעדכן איזשהו מרחק בשלב שמספרו כמספר צמתי הרכיב ועוד 1, יש מסלול למעגל שלילי: יש מעגל שלילי ברכיב ולגבי כל צומת שברכיב יש מסלול למעגל. נחשב את גרף העל של רכיבי הקשירות החזקה. נהפוך את כל כיווני הקשתות של גרף זה ונוסיף צומת שממנו יצאו קשתות לכל הרכיבים שבהם התגלה מעגל שלילי. כל צומת שלגביו יש מסלול מהצומת שהוספנו אליו, יש לו מסלול למעגל שלילי בגרף המקורי.

<u>שאלה 76</u> **(מרצים: פרופ' רון שמיר, פרופ' יוסי עזר, מתרגלים: טל ינקוביץ', ג'אד סלבאק)**
נתון אוסף של n נקודות (x_i, y_i) $i = 1, \dots, n$ על המישור. עבור פונקציה $f: R \to R$ ונקודה $p_i = (x_i, y_i)$, הסטייה של p_i מ- f מוגדרת בתור הערך הבא: $y_i - f(x_i)$. נסחו כבעיות תכנון לינארי את שתי הבעיות הבאות:

<u>א</u>. **הסטייה הכוללת** של הנקודות מהפונקציה מודגרת **כסכום הערכים המוחלטים** של הסטיות של כל n הנקודות. יש למצוא ישר $f(x) = ax + b$ עם סטייה כוללת מינמלית.

<u>ב</u>. **הסטייה המקסימלית** של הנקודות מהפונקציה מוגדרת **כמקסימום הערך המוחלט** של סטייה בנקודה כלשהיא. יש למצוא פרבולה $g(x) = ax^2 + bx + c$ עם סטייה מקסימלית קטנה ככל האפשר.

פורמולציה והסבר סעיף א':
נגדיר משתנים אי שליליים $a_p, a_{neg}, b_p, b_{neg}$ ומשתנים אי שליליים d_i עבור כל נקודה i. ניתן לבטא את השיפוע כ $a_p - a_{neg}$ ואת הקבוע שבפונקציה כ $b_p - b_{neg}$. קבוצת האילוצים תהיה עבור כל i :

$$d_i \geq (a_p - a_{neg})x_i + (b_p - b_{neg}) - y_i$$
$$d_i \geq -[(a_p - a_{neg}\ )\ x_i + (b_p - b_{neg}\ ) - y_i]$$

פונקצית המטרה תהיה $min \sum_{i=1}^{n} d_i$.
מכיון שמחפשים ערך מינימום אז המשתנים d_i יקבלו בדיוק את הערכים המוחלטים ולא יותר.

פורמולציה והסבר סעיף ב':
נגדיר משתנים אי שליליים $a_p, a_{neg}, b_p, b_{neg}, c_p, c_{neg}$ ומשתנה אי שלילי d.
קבוצת האילוצים תהיה עבור כל i:

$$d \geq (a_p - a_{neg})x_i{}^2 + (b_p - b_{neg})x_i + (c_p - c_{neg}) - y_i$$
$$d \geq -[(a_p - a_{neg}\ )x_i{}^2\ + (b_p - b_{neg})x_i + (c_p - c_{neg}) - y_i]$$

פונקצית המטרה תהיה $min\{d\}$.

<u>שאלה 77</u> (מרצים: פרופ' נוגה אלון, ד"ר רני הוד, מתרגלים: אופיר פרידלר, אלון עדן)

<u>א</u>. הוכיחו כי בכל גרף דו-צדדי 4-רגולרי (ז"א, דרגת כל צומת בו היא 4) יש זיווג מושלם.

<u>ב</u>. תארו אלגוריתם יעיל ככל האפשר המוצא זיווג מושלם בגרף דו-צדדי 4-רגולרי.

<u>יעילות</u>: $O(|V|)$

<u>אלגוריתם והסבר</u>:

בכל רכיב קשירות נמצא מעגל אוילר (מכיון שכל הדרגות זוגיות אז קיים מעגל אוילר בכל רכיב קשירות). נעבור על המעגל ונמחק כל קשת שניה. מכיון שהגרף דו צדדי אז לא קיים מעגל באורך אי זוגי ולכן לכל צומת נחזור לאחר מעבר על מספר זוגי של קשתות. לכן לאחר המחיקה דרגת כל צומת תהיה 2. כעת נחזור על תהליך דומה ובכל מעגל נמחק כל קשת שניה. לאחר מחיקה זו נשאר עם גרף 1-רגולרי. זהו למעשה זיווג מושלם. בגרף המקורי לכל צומת יש דרגה 4, לכן סכום הדרגות הוא $4|V|$ ומתקיים $|E| = |2V|$.

<u>שאלה 78</u> (מרצה: פרופ' מיכה שריר, מתרגלים: רני הוד, שי ורדי)

נתונה רשת זרימה $G = (V, E)$ עם קיבולים $c: E \to R^+$, מקור s ובור t. תארו אלגוריתם יעיל שמחשב, בהנתן זוג קשתות $e_1, e_2 \in E$, האם <u>קיים</u> חתך מינימלי ברשת שמכיל <u>בדיוק אחת</u> משתי הקשתות e_1, e_2.

<u>יעילות</u>: של אלגוריתם לחישוב זרימה מכסימלית

<u>אלגוריתם והסבר</u>:

נחשב זרימה מכסימלית ברשת.

נבחר גודל חיובי A שקטן מקיבולי כל אחת משתי הקשתות.

נריץ אלגוריתם לחישוב זרימה מכסימלית בשתי רשתות שונות. בכל אחת מהן הקיבול של קשת אחת מהשתיים קטן ב A ביחס לקיבולה המקורי והקיבול של האחרת גדול ב A ביחס לקיבול המקורי שלה. קיבולי יתר הקשתות יהיו ללא שינוי.

אם"ם בלפחות אחת משתי ההרצות הזרימה תהיה קטנה ב A ביחס לזרימה המקורית, אז קיים חתך מינימלי שמכיל את בדיוק אחת משתי הקשתות האלה.

<u>הסבר</u>

הקטנת קיבול של קשת בגודל A מקטינה את גודל הזרימה בגודל זה רק אם קשת זו היא בחתך מינימלי. אם היא לא נמצאת בחתך מינימלי אז הזרימה יכולה לקטון אבל בפחות. אם הקשת האחרת נמצאת בכל חתך מינימלי שבו הראשונה נמצאת, אז גדלי החתכים המינימלים שבהם הקשת שהקטנו את קיבולה לא יקטנו, וחתכים אחרים הם גדולים יותר, ולכן כל גדלי החתכים יהיו גדולים מאשר גודל הזרימה ברשת המקורית פחות A. אם יש חתך מינימלי שבו הראשונה נמצאת והשניה לא נמצאת, אז גודל חתך זה יקטן ב A.

<u>שאלה 79</u> (מרצים: פרופ' אורי צוויק, ד''ר תומר קורן, מתרגלים: טל ינקוביץ', ג'אד סלבאק)
נתון גרף מכוון $G = (V, E)$ ו- $n = |V|$ פונקציות משקל $w_1, w_2, \cdots, w_n : E \to R$ על קשתותיו.
(משקלי הקשתות יכולים להיות חיוביים או שליליים ולא ניתן להניח שום דבר לגביהם.)
המשקל של מסלול $P = (u_0, u_1, \cdots, u_k)$ כאשר $k \leq n$, מוגדר להיות $\sum_{i=1}^{k} w_i(u_{i-1}, u_i)$,
כלומר המשקל של הקשת ה-i-ית על המסלול נקבע על פי פונקצית המשקל w_i. משקל מסלול בעל יותר מ-n קשתות אינו מוגדר.

<u>סעיף א'</u>
תארו אלגוריתם יעיל ככל האפשר שבהינתן צומת s מוצא עבור כל צומת v בגרף מסלול מ-s ל-v (עם לכל היותר n קשתות) בעל משקל מינימלי, אם יש כזה. המסלול לא חייב להיות פשוט, כלומר מותר לבקר בצומת יותר מפעם אחת. על האלגוריתם להחזיר תיאור של מסלולים אלה, לא רק את משקלם. הוכיחו את נכונות האלגוריתם ונתחו את סיבוכיותו.

<u>יעילות</u>: $O(|V||E|)$

<u>אלגוריתם והסבר</u>:
נבנה גרף של $n + 1$ שכבות. בשכבה הראשונה יהיה רק הצומת s וביתר השכבות יהיו כל צמתי הגרף. בין כל נציג של צומת בשכבה אחת לנציג של אותו צומת בשכבה העוקבת תהיה קשת אם"ם בגרף המקורי יש קשת בין הצמתים המקוריים. משקלי הקשתות שבין שכבה k לשכבה $k + 1$ יהיו לפי פונקצית המשקל ה-k. נחשב את המרחקים לצמתי כל שכבה לאחר שכבר יהיו לנו המרחקים לשכבה הקודמת. המרחק לצומת יהיה שווה למינימום על פני סכום המרחקים לאבות האפשריים שלו בתוספת המרחק מהאב האפשרי הזה אליו. אם המרחקים לשכבה הקודמת הם הקצרים ביותר, אז נקבל על-ידי מיצוי כל האפשרויות, את המרחקים האופטימלים לשכבה זו. אחר-כך לגבי כל צומת נמצא את הנציג שלו שהמסלול אליו מ-s הכי זול. לגבי כל צומת בכל שכבה נשמור אב שדרכו המסלול קל ביותר. כך נוכל לקבל תיאור של מסלולים אופטימלים. על כל קשת עוברים ב $|V|$ המעברים משכבה לשכבה ובמציאת האבות של הצמתים בגרף העזר. לגבי כל צומת מקורי יש $|V|$ פעולות למציאת נציגו בעל מרחק מינימלי.

<u>סעיף ב'</u>
תארו אלגוריתם יעל ככל האפשר שבהינתן צומת t מוצא עבור כל צומת אחר v בגרף מסלול מ-v ל-t (עם לכל היותר n קשתות) בעל משקל מינימלי, אם יש כזה. על האלגוריתם להחזיר תיאור של מסלולים אלה, לא רק את משקלם. הוכיחו את נכונות האלגוריתם. מה סיבוכיות האלגוריתם ?

<u>יעילות</u>: $O(|V||E|)$

<u>אלגוריתם והסבר</u>:
נשתמש בתכנות דינאמי. לגבי כל צומת v נחשב את אורכם של חלקי המסלולים ממנו לצומת t כאשר מיקומו במסלול הוא k שהוא כל גודל אפשרי עד n. נחשב את הפתרונות עבור $P_{v,k}$ שבה שני פרמטרים. $P_{v,k}$ יהיה שווה למינימום על פני הצמתים u של הסכום של משקל הקשת שבין v ל u לפי פונקצית המשקל ה-k-ית ועוד $P_{u,k+1}$. תנאי ההתחלה הם $P_{t,k} = 0$ ועבור כל צומת אחר $P_{v,n} = \infty$.
לבסוף עבור כל צומת נמצא את אורך המסלול כשהוא במקום הראשון. על כל קשת עוברים $|V|$ פעמים.

שלומי