<u>ממ"ן 13</u>

שאלה 1

א.

יהי p מסלול בין s לבין t שמכיל רק קשתות שימושיות. נראה ש p הוא בהכרח מסלול קצר ביותר.

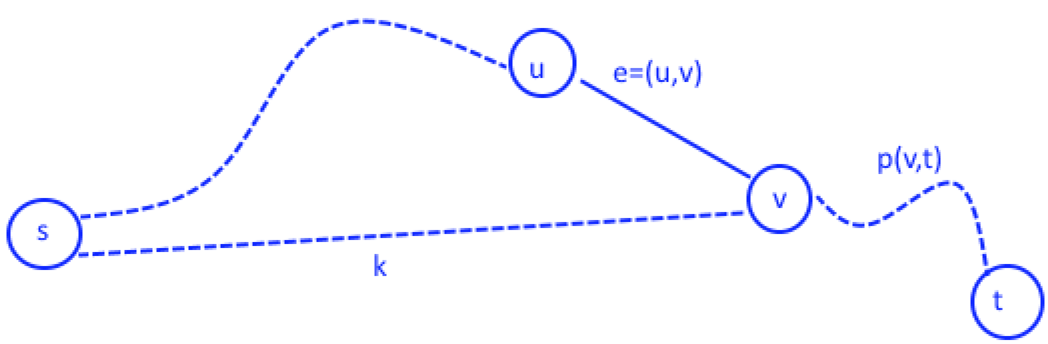
נוכיח באינדוקציה על האורך (מספר הקשתות) n של המסלול p:
עבור n=1 ברור שהטענה מתקיימת כי הקשת היחידה במסלול היא שימושית ולכן שייכת למסלול קצר ביותר בין s ל t אבל הקשת מהווה את המסלול כולו ולכן המסלול חייב להיות קצר ביותר.
נניח שהטענה נכונה עבור כל מסלול באורך n=N שכל קשתותיו הן קשתות שימושיות ונוכיח שהיא מתקיימת עבור כל מסלול באורך n=N+1 שכל קשתותיו הן קשתות שימושיות:
יהי מסלול p באורך N+1. אז אפשר לחלק אותו לשני חלקים:

- החלק של המסלול באורך N מהקודקוד s לקודקוד v1 שנמצא לפני הקודקוד t במסלול p. נסמן חלק זה כתת מסלול q.
- הקשת שמחברת את הקודקוד v1 ל t במסלול p.

לפי הנחת האינדוקציה נקבל שהמסלול q בין s לבין v1 הוא מסלול קצר ביותר בין s לבין v1. כמו כן נתון שהקשת בין v1 לבין t היא שימושית ולכן היא הקשת האחרונה במסלול קצר ביותר בין s לבין t. אם נניח בשלילה שהמסלול p אינו מסלול קצר ביותר בין s לבין t אז נקבל שחייב להיות מסלול r בין s לבין v1 שמשקלו קטן יותר ממשקלו של q (כי הקשת בין v1 ל t היא שימושית) בסתירה לכך ש q הוא מסלול קצר ביותר בין s ל v1. הנחת השלילה הובילה לסתירה ולכן המסלול p הוא מסלול קצר ביותר בין s ל t.
מ.ש.ל.

ב.

נניח שקיימת קשת $e=(u,v)$ במסלול כלשהו p מ s ל t שאינה קשת שימושית. לכן בהכרח קיים מסלול קצר יותר k בין s ל v שמשקלו קטן יותר מהמשקל של החלק של p שמתחיל ב s ונגמר ב v (אם לא קיים מסלול קצר יותר בין s ל v אז החלק של p בין s ל v הוא מסלול קצר ביותר בין s ל v ולכן הקשת e היא שימושית בסתירה להנחתנו שקשת זו אינה קשת שימושית).
מכאן שהמסלול שהוא איחוד של k עם החלק של המסלול ב p שמתחיל ב v ומסתיים ב t הוא קצר יותר מ p ולכן p אינו מסלול קצר ביותר בין s ל t (ר' איור)



הסבר: מכיוון ש e אינה קשת שימושית אז החלק של המסלול ב p בין s ל v אינו מסלול קצר ביותר בין s ל v ולכן אפשר להחליפו במסלול קצר יותר k בין s ל v ולקבל מסלול קצר יותר בין s ל t שמורכב מ k ומחלק המסלול ב p בין v ל t (החלק $p(v,t)$).

ג.

יהי p מסלול שני קצר ביותר מ s ל t. המסלול p אינו מסלול קצר ביותר בין s ל t ולכן חייבת להיות בו לפחות קשת לא שימושית אחת לפי טענה א.

נניח בשלילה שקיימות ב p לפחות שתי קשתות לא שימושיות ונוכיח שבמקרה זה p אינו מסלול שני קצר ביותר בין s ל t.

נבחר שתי קשתות לא שימושיות ב p (ייתכן שיש יותר) ונסמן אותן $e1 = (u, v),\ e2 = (x, z)$. נניח בלי הגבלת הכלליות שהקשת e1 קודמת לקשת e2 במסלול מ s ל t.

מהעובדה שהקשת e1 לא שימושית נובע שקיים מסלול k1 מ s ל v שהינו קצר יותר מהתת-מסלול של p בין s ל v. נסמן ב f את המסלול שמורכב מהמסלול k1 והתת-מסלול של p שמתחיל ב v ומסתיים ב t. מהבנייה נובע שמתקיים:

$$weight(f) < weight(p)$$

מכך ש e2 מופיעה ב p אחרי הקשת e1 נובע ש e2 נמצאת במסלול f. אבל e2 לא שימושית ולכן קיים מסלול k2 בין s לבין z שהוא קצר יותר מהתת-מסלול של p בין s ל z. נסמן ב g את המסלול שמורכב מהמסלול k2 והתת-מסלול של p שמתחיל ב z ומסתיים ב t. מהבנייה נובע שמתקיים:

$$weight(g) < weight(f)$$

ובסך הכל מתקיים:

$weight(g) < weight(f) < weight(p)$ בסתירה לכך שהמסלול p הוא מסלול שני קצר ביותר מ s ל t. הנחת השלילה הובילה לסתירה ולכן אין 2 או יותר קשתות לא שימושיות ב p אבל מטענה א נובע שקיימת לפחות קשת לא שימושית אחת ב p ולכן בהכרח קיימת ב p בדיוק קשת לא שימושית אחת.

אם נסמן קשת זו ב $e = (u, v)$ אז בהכרח מתקיים שהרישא של p מ s ל u הוא מסלול קצר ביותר מ s ל u. אם נניח בשלילה שלא כך הוא - נקבל שהרישא של p מ s ל u חייבת לכלול לפחות קשת לא שימושית אחת (לפי טענה א) אבל אז נקבל שב p יש לפחות שתי קשתות לא שימושיות בסתירה לכך שב p יש בדיוק קשת לא שימושית אחת.

נשתמש באותו טיעון בדיוק כדי להוכיח שהסיפא של p מ v ל t היא מסלול קצר ביותר מ v ל t: אם נניח בשלילה שהסיפא של p מ v ל t אינה מסלול קצר ביותר מ v ל t נקבל שהסיפא חייבת לכלול לפחות קשת לא שימושית אחת אבל אז נקבל שב p יש לפחות שתי קשתות לא שימושיות בסתירה לכך שב p יש בדיוק קשת לא שימושית אחת.

ד.

Algorithm

The algorithm is composed from a main section and a recursive procedure that is used to evaluate candidate paths.

MAIN ALGORITHM

1. Perform Dijkstra algorithm on G to find shortest distances from s to all other vertices in G. Once Dijkstra ends we'll have the distance of every vertex u from s in the d[u] array entry.
2. Initialize $MIN = \infty$
3. Initialize VISITED array to FALSE (contains an entry per vertex)
4. Call **EVAL_CANDIDATES** and pass vertex t and w=0 as parameters
5. IF MIN = ∞ THEN
    a. DECLARE "No second shortest path exists"
6. ELSE
    a. DECLARE "second shortest path has {MIN} weight"
7. END IF

END MAIN ALGORITHM

PROCEDURE EVAL_CANDIDATES(vertex n, weight w)

1. VISITED[n] = TRUE
2. FOR EVERY EDGE e=(u,n) whose tail is vertex u and head is vertex n DO
    a. IF d[u]+weight(e) > d[n] THEN
        i. IF d[u]+weight(e)+w < MIN THEN
            1. $MIN = d[u] + weight(e) + w$
        ii. END IF
    b. ELSE IF VISITED[u] = FALSE AND $u \neq s$ THEN
        i. EVAL_CANDIDATES(u, w+weight(e))
    c. END IF
3. END FOR

END EVAL_CANDIDATES

הוכחת נכונות:

טענת עזר: אם בעת ביצוע האלגוריתם מתקיים התנאי (a):

IF d[u]+weight(e) > d[n]

אז הקשת e אינה שימושית, אחרת e היא קשת שימושית.

הוכחה: הקשת e הינה הקשת האחרונה במסלול בין s ל n. מסלול זה נראה כך: $s \to ... \to u \to n$

אבל לפי התנאי (a) מתקיים שמשקל המסלול שמורכב ממסלול קצר ביותר בין s ל u ועוד הקשת e גדול ממשקל מסלול קצר ביותר בין s לבין n. לכן לא ייתכן ש e היא הקשת האחרונה במסלול קצר ביותר בין s לבין n ומכאן ש e אינה שימושית.
אם התנאי (a) לא מתקיים אז המשקל של מסלול קצר ביותר בין s ל u ועוד המשקל של e זהה למשקל של מסלול קצר ביותר בין s ל n ולכן גם האיחוד של מסלול קצר ביותר בין s ל u והקשת e מהווה מסלול קצר ביותר בין s ל n ומכאן ש e היא הקשת האחרונה במסלול קצר ביותר בין s ל n ולכן היא קשת שימושית.

כעת נוכיח שהמסלול שמופק ע"י האלגוריתם לעיל הוא מסלול שני קצר ביותר. כדי להוכיח זאת מספיק להוכיח שיש במסלול זה בדיוק קשת אחת לא שימושית (לפי טענה ג').

יש שתי אפשרויות:

1. אין מסלול שני קצר ביותר ב G
2. יש מסלול שני קצר ביותר ב G

במקרה (1) הבדיקה אם המשקל של מסלול קצר ביותר בין s ל u בצירוף משקל הקשת e תמיד תיכשל (אחרת לפי טענת העזר e היא קשת לא שימושית והצירוף שלה עם מסלול קצר ביותר בין s לבין u יוצר מסלול שני קצר ביותר בסתירה לכך שאין מסלול שני קצר ביותר ב G) ולכן בסופם של הקריאות הרקורסיביות יישאר ב MIN הערך אינסוף - והמשמעות היא שאין מסלול שני קצר ביותר כנדרש.

במקרה (2) חייב להיות מסלול בין s ל t שבו יש בדיוק קשת לא שימושית אחת (לפי טענה ג').
לכן המסלול נראה כך: $s \to ...u \to n \to ... \to t$ (הקשת בין u ל n היא הקשת הלא שימושית היחידה במסלול).
נראה שהאלגוריתם חייב לכלול את המסלול הזה בבדיקת המשקל המינימלי.
כל הקשתות בין n ל t הן קשתות שימושיות ולכן האלגוריתם יסרוק את כולן במסגרת הקריאות הרקורסיביות של EVAL_CANDIDATES. במהלך בדיקת הקשת $e=(u,n)$ האלגוריתם יגלה שקשת זו אינה שימושית ובנקודה זו יבחן את משקל המסלול שמורכב ממסלול קצר ביותר מ s ל u ועוד משקל הקשת e ועוד משקל הקשתות שבהן ביקרנו עד כה במסלול מ n אל t. שים לב שמסלול זה הוא מועמד למסלול שני קצר ביותר כי יש בו בדיוק קשת לא שימושית אחת והוא מחבר את s ל t. אם משקלו קטן יותר ממשקל המסלול האחרון שבדקנו אז שומרים את משקלו כמינימום החדש ובאופן זה מבטיחים שבסיום האלגוריתם יוחזר משקל המסלול המינימלי.
מכך שהאלגוריתם בודק את כל המסלולים עם בדיוק קשת לא שימושית אחת ומכך שהוא מתחזק את המשקל המינימלי בין כל מסלולים אלה נובע שהאלגוריתם חייב להחזיר את משקל המסלול המינימלי מבין כל המסלולים המועמדים להיות מסלולים שניים קצרים ביותר ולכן בהכרח האלגוריתם יחזיר את משקל מסלול שני קצר ביותר.

<u>סיבוכיות האלגוריתם:</u>
סיבוכיות האלגוריתם נגזרת מחלקיו:

1. הרצת אלגוריתם דייקסטרה תורמת סיבוכיות של $O(mlogn)$ כאשר m הוא מספר הקשתות ב G ו n הוא מספר הקודקודים ב G.
2. אתחול מערך VISITED תורם סיבוכיות של $O(n)$

3. הרצת הפרוצדורה הרקורסיבית EVAL_CANDIDATES תורמת לכל היותר $O(m)$ כי כל קשת נבדקת פעם אחת לכל היותר (זאת מכיוון שאנחנו מסמנים כל קודקוד שבו ביקרנו ונמנעים להיכנס שנית לאותו קודקוד ולכן לאותה קשת).

לכן הסיבוכיות הכוללת של האלגוריתם היא $O(n+mlogn)$ כי הסיבוכיות $O(m)$ לא תורמת בחישוב האסימפטוטי. אם G הוא גרף קשיר אז הסיבוכיות הכוללת היא $O(mlogn)$ כי יש לפחות n-1 קשתות ב G ואז הסיבוכיות $O(n)$ הופכת ללא משמעותית ביחס ל $O(mlogn)$.

שאלה 2

נתון ש:

$$DFT_n(p(x))=(v_1,\dots v_n)$$

לכן מתקיים:

$$p(\omega_{0,n})=v_1$$
$$p(\omega_{1,n})=v_2$$
$$\dots$$
$$p(\omega_{n-1,n})=v_n$$

כאשר $0\leq k<n \qquad \omega_{k,n}=e^{\frac{-2\pi ik}{n}}$

מהגדרת השורש לעיל מתקיים:

$$\omega_{k,n}=e^{\frac{-2\pi ik}{n}}=\left(e^{\frac{-2\pi ik}{2n}}\right)^2=(\omega_{k,2n})^2$$

מהגדרת ה DFT מתקיים:

$$DFT_{2n}(p(x^2))=(\ p(\omega_{0,2n}{}^2),\ p(\omega_{1,2n}{}^2),\ \dots\ ,\ p(\omega_{n-1,2n}{}^2),\ p(\omega_{n,2n}{}^2),\ p(\omega_{n+1,2n}{}^2),\ \dots\ p(\omega_{2n-1,2n}{}^2)\ )$$

ולכן:

$$DFT_{2n}(p(x^2))=(p(\omega_{0,n}),\ p(\omega_{1,n}),\ \dots\ p(\omega_{n-1,n}),\ p(\omega_{0,n}),\ p(\omega_{1,n}),\ \dots\ p(\omega_{n-1,n}))$$

כלומר - התוצאה בעצם מורכבת מהתמרת פוריה המקורית כשכמות האיברים מוכפלת.

העובדה שסידרת האיברים המקורית מופיעה פעמיים בתוצאה נובעת מהשיוויון הבא:

$$\omega_{k+n,n}=e^{\frac{-2\pi i(k+n)}{n}}=e^{\frac{-2\pi ik}{n}-2\pi i}=\frac{e^{\frac{-2\pi ik}{n}}}{e^{2\pi i}}=e^{\frac{-2\pi ik}{n}}=\omega_{k,n} \qquad 0\leq k<n$$

לפיכך נקבל לבסוף:

$$DFT_{2n}(p(x^2))=(v_1,\dots v_n,v_1,\dots v_n)$$

שאלה 3

א.

נסמן את המטריצה A מחדש כ M כדי לא ליצור בלבול עם הסימונים למטה.
נכתוב את המטריצה M באופן הבא:

$$M^2=\begin{pmatrix} a_0 & a_1 \\ b_0 & b_1 \end{pmatrix}\times\begin{pmatrix} a_0 & a_1 \\ b_0 & b_1 \end{pmatrix}=\begin{pmatrix} a_0a_0+a_1b_0 & a_0a_1+a_1b_1 \\ b_0a_0+b_1b_0 & b_0a_1+b_1b_1 \end{pmatrix}$$

אז מתקיים:

$$\begin{aligned}
A&=(a_0+a_1)\cdot(a_0+b_0)=a_0a_0+a_0b_0+a_1a_0+a_1b_0\\
B&=(a_1+b_1)\cdot(b_1+b_0)=a_1b_1+a_1b_0+b_1b_1+b_1b_0\\
C&=(b_0+a_1)\cdot a_0=a_0b_0+a_0a_1\\
D&=(a_1+b_0)\cdot b_1=a_1b_1+b_0b_1\\
E&=(a_0+b_1)\cdot a_1=a_1a_0+a_1b_1
\end{aligned}$$

מכאן קל לראות שמתקיים:

$$\begin{aligned}
&A-C=a_0a_0+a_0b_0+a_1a_0+a_1b_0-a_0b_0-a_0a_1=a_0a_0+a_1b_0=[M^2]_{0,0}\\
&B-D=a_1b_1+a_1b_0+b_1b_1+b_1b_0-a_1b_1-b_0b_1=a_1b_0+b_1b_1=[M^2]_{1,1}\\
&C+D-E\ =\ a_0b_0+a_0a_1+a_1b_1+b_0b_1-a_0a_1-a_1b_1=a_0b_0+b_0b_1=[M^2]_{1,0}\\
&E=a_0a_1+a_1b_1=[M^2]_{0,1}
\end{aligned}$$

באמצעות 5 פעולות כפל קיבלנו את $A^2=M^2$ כמבוקש.

ב.

אין אפשרות להשתמש בגישה רקורסיבית שמבוססת על סעיף א' מכיוון שלאחר פירוק הבעיה בגודל n ל 5 תת בעיות בגודל n/2 מקבלים מכפלות שאינן ריבועיות ולכן אין אפשרות להמשיך את הרקורסיה.

לדוגמא, אם נניח ש M מטריצה ריבועית עם n איברים (נניח ש n הוא חזקה שלמה של 2 לצורך הפשטות) ונחלק את M ל 4 מטריצות בגודל n/2 אז נקבל:

$$M^2=\begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \times \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix}$$

כעת נמשיך כפי שעשינו בסעיף א' וניצור מטריצות בגודל n/2 כדי לקבל 5 מכפלות:

$$\begin{aligned} A &= (M_{1,1}+M_{1,2})\times(M_{1,1}+M_{2,1}) \\ B &= (M_{1,2}+M_{2,2})\times(M_{2,2}+M_{2,1}) \\ C &= (M_{2,1}+M_{1,2})\times M_{1,1} \\ D &= (M_{1,2}+M_{2,1})\times M_{2,2} \\ E &= (M_{1,1}+M_{2,2})\times M_{1,2} \end{aligned}$$

כעת אנחנו בבעיה כי המטריצות שמרכיבות את המכפלות אינן בהכרח זהות ולכן הבעיה הופכת מבעיה של למצוא ריבוע של מטריצה לבעיה של כפל מטריצות רגיל ואין אפשרות להמשיך ברקורסיה (לפחות לא לפי השיטה שהוצגה בסעיף א').

אם אפשר היה להשתמש בגישה רקורסיבית שמבוססת על א' אז היינו מקבלים את יחס הנסיגה:

$$T(n)=5\cdot T\left(\frac{n}{2}\right)+k\cdot n^2$$

הערה: k הוא קבוע שתלוי במספר פעולות חיבור/חיסור מטריצות שאנחנו עושים בכל פעם שאנו מפצלים מטריצה ל 5 תת מטריצות ועושים את החישובים שהוצגו בסעיף א' (בהבדל אחד שפה מדובר על חיבור מטריצות). מכיוון שחיבור מטריצות דורש מעבר על n^2 איברים ומכיוון שאנחנו מפצלים מטריצה בגודל n ל 5 מטריצות קטנות יותר בגודל n/2 כ"א אנחנו מקבלים את יחס הנסיגה לעיל.

פריסה של עץ הרקורסיה תוביל לביטוי הבא:

$$k\cdot n^2\cdot\sum_{d=0}^{\log_2 n}5^d\cdot\left(\frac{1}{2^d}\right)^2=k\cdot n^2\cdot\left(\frac{\left(\frac{5}{4}\right)^{1+\log_2 n}-1}{\frac{5}{4}-1}\right)=k\cdot n^2\cdot 4\cdot\left(\frac{5}{4}\cdot\frac{n^{\log_2 5}}{n^{\log_2 4}}-1\right)$$

$$=k\cdot 5\cdot n^{\log_2 5}-k\cdot n^2\cdot 4\ <\ k\cdot 5\cdot n^{\log_2 5}$$

ומכאן שמתקיים:

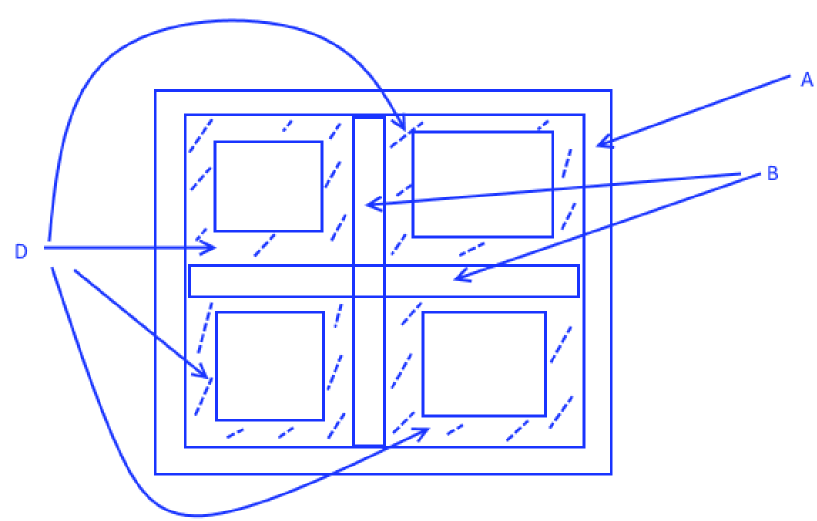
$$T(n)=O(n^{\log_2 5})$$

כלומר - אם אפשר היה להשתמש ברקורסיה באמצעות השיטה שהוצגה בסעיף א' אז הפרופסור היה צודק בטענתו לגבי סיבוכיות האלגוריתם שהציע.

שאלה 4

נגדיר את הקבוצות הבאות:
הקבוצה A תהיה כל הקודקודים שנמצאים בגבול של G (כל הקודקודים עם פחות מ 4 קשתות בסריג G).
הקבוצה B תהיה כל הקודקודים שנמצאים בשורה האמצעית של הסריג G ובנוסף כל הקודקודים שנמצאים בעמודה האמצעית של הסריג G (לא כולל קודקודים ב A).
הקבוצה C תהיה כל הקודקודים שנמצאים ב A ועוד הקודקודים ב B.
הקבוצה D תהיה כל הקודקודים שסמוכים לקודקודים שנמצאים ב C.

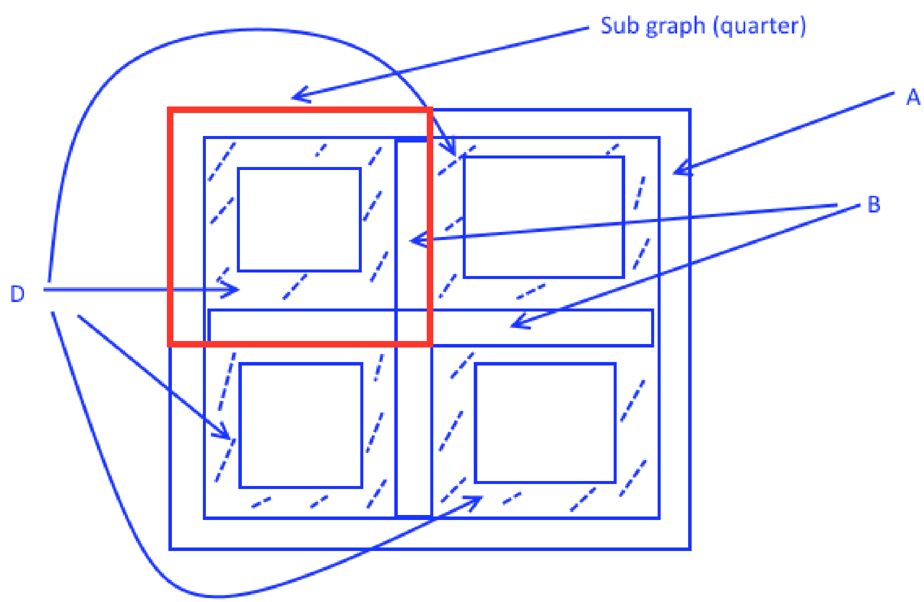
איור:



הערה: הקבוצה D מתוארת כשטח עם הקווים המקווקוים באיור לעיל.

האלגוריתם יוגדר באופן רקורסיבי על סריג בגודל n x n:

1. אם גודל הסריג קטן מ 4x4 בצע סריקה מלאה על כל קודקודי הסריג כדי למצוא מינימום מקומי.
2. אחרת - בצע סריקה של כל הקודקודים ב A כדי למצוא את הקודקוד עם התווית המינימלית מבין הקודקודים ב A. סמן את הקודקוד שנמצא ב w.
3. אם w נמצא באחת מפינות הסריג (יש לו בדיוק שתי קשתות ב G) אז מצאנו מינימום מקומי והאלגוריתם מסיים.
4. אחרת w לא נמצא בפינה ולכן יש לו קשת אמצעית שמתחברת לקודקוד פנימי של G שלא שייך ל A. נסמן קודקוד זה כ z. אם התווית של z גדולה מזו של w אז בהכרח w הוא מינימום מקומי ב G והאלגוריתם מסיים.
5. אחרת בצע סריקה של כל הקודקודים שנמצאים ב D ו B כדי למצוא את הקודקוד עם התווית המינימלית מבין הקודקודים ב D ו B. סמן את הקודקוד שנמצא עם התווית המינימלית כ v.
6. אם הקודקוד v שייך לקבוצה B אז הוא בהכרח מינימום מקומי ב G והאלגוריתם מסיים (ר' הסבר בהמשך).
7. אם הקודקוד v לא שייך לקבוצה B אז הוא בהכרח נמצא בקבוצה D. כעת נגדיר תת-סריג של G כרביע של G (הרביע שמכיל את הקודקוד v) וכן את קודקודי הגבול של G באותו רביע וקודקודי הקבוצה B ששייכים לאותו רביע. ר' איור:



כעת נקרא לאלגוריתם באופן רקורסיבי על התת-סריג שהגדרנו ונחזיר את המינימום המקומי שיחושב ע"י הקריאה הרקורסיבית.

<u>הוכחת נכונות:</u>

הנכונות של סעיפים 1,2,3,4 ברורה.

לגבי סעיף 6: אם הקודקוד v שייך לקבוצה B אז הוא חייב להיות מינימום מקומי ב G כי הוא מוקף בקודקודים ששייכים ל B, ל D או A. ברור שהוא קטן יותר מאלו של B ו D והוא גם חייב להיות קטן יותר מאלו שב A כי יש כבר בקבוצה $B \bigcup D$ את הקודקוד z שקטן יותר מכל הקודקודים ב A והקודקוד v הוא שווה או קטן ל z (הוא הקודקוד המינימלי ב $B \bigcup D$).

לגבי סעיף 7: הקריאה הרקורסיבית על הרביע שנבחר חייבת למצוא מינימום מקומי בסופו של דבר עקב הגעה לתחתית הרקורסיה (חייב להיות מינימום מקומי כי חייב להיות קודקוד מינימלי בכל קבוצה נבחרת של קודקודים כי התויות של כל הקודקודים שונות).

הערה חשובה:
במקרה שהאלגוריתם מגלה מינימום המקומי בתוך הקריאה הרקורסיבית (על תת-סריג) המינימום המקומי חייב להיות גם מינימום מקומי של G כולו. זאת עקב העבודה שהקריאה הרקורסיבית מתבצעת רק אחרי שהוברר כבר שהמינימום המקומי לא נמצא בקודקוד גבול של התת-סריג.

<u>ניתוח סיבוכיות:</u>

כדי לפשט את הניתוח נתאר את הסיבוכיות כפונקציה של n שהוא מספר הקודקודים בצלע של הסריג. כלומר - מספר הקודקודים בסריג הוא n x n.
נסמן את הסיבוכיות של סריקת הקודקודים של A ב TA

נסמן את הסיבוכיות של סריקת הקודקודים של $B \bigcup D$ ב TDB

אז קל לראות שמתקיים:

$$TA(n) \simeq 4n \Rightarrow TA(n) = O(n)$$

$$TDB(n) \simeq 4 \cdot 4 \cdot \frac{n}{2} + 2n \Rightarrow TDB(n) = O(n)$$

בנוסף מתבצעת קריאה רקורסיבית על תת-סריג שאורך צלעו n/2 מאורך צלע הסריג המקורי. לכן נקבל בסה"כ את יחס הנסיגה:

$$T(n) = T\left(\frac{n}{2}\right) + O(n)$$

נפרוס את עץ הרקורסיה ונקבל:

$$k \cdot n \cdot \sum_{d=0}^{\log_2 n} \left(\frac{1}{2}\right)^d = k \cdot n \cdot \frac{\left(\frac{1}{2}\right)^{1+\log_2 n} - 1}{\frac{1}{2} - 1} = k \cdot n \cdot 2 - k < k \cdot n \cdot 2$$

כאשר kn מייצג את הסיבוכיות של הסריקות על A ועל $B \bigcup D$.

מכך אפשר לראות שמתקיים: $T(n) = O(n)$ כפי שנדרש בשאלה.